

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

## ⑫ 公開特許公報(A) 平2-187898

⑤ Int.Cl.<sup>5</sup>

識別記号

庁内整理番号

⑬ 公開 平成2年(1990)7月24日

G 08 G 1/0969  
G 01 C 21/00

N

6821-5H  
6752-2F

審査請求 未請求 請求項の数 2 (全32頁)

⑭ 発明の名称 ナビゲーション装置の位置入力方式

⑯ 特 願 平1-7999

⑰ 出 願 平1(1989)1月17日

⑱ 発 明 者 諸 戸 脩 三 愛知県安城市藤井町高根10番地 アイシン・エイ・ダブリ  
ユ株式会社内⑲ 発 明 者 小 林 康 二 愛知県安城市藤井町高根10番地 アイシン・エイ・ダブリ  
ユ株式会社内⑳ 発 明 者 角 谷 孝 二 愛知県安城市藤井町高根10番地 アイシン・エイ・ダブリ  
ユ株式会社内㉑ 出 願 人 アイシン・エイ・ダブリ 愛知県安城市藤井町高根10番地  
リュ株式会社

㉒ 出 願 人 株式会社新産業開発 東京都渋谷区幡ヶ谷1丁目番33番3号

㉓ 代 理 人 弁理士 阿部 龍吉 外5名

最終頁に続く

## 明 細 書

## 1. 発明の名称

ナビゲーション装置の位置入力方式

## 2. 特許請求の範囲

(1) 指定された出発地と目的地を基に走行ルートを設定し、走行案内を行うナビゲーション装置において、複数の位置に関する情報からなる位置データ及び該位置データの格納情報と電話番号とを対応させるリストを有し、出発地や目的地等の位置を指定するデータとして電話番号を入力することにより位置データを読み出し可能にしたことを特徴とするナビゲーション装置の位置入力方式。

(2) リストに特定の情報を有し、入力された電話番号が登録されていない場合には特定の情報を与えるようにしたことを特徴とする請求項1記載のナビゲーション装置の位置入力方式。

## 3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は、指定された出発地と目的地を基に走行ルートを設定し、走行案内を行うナビゲーション

ン装置の位置入力方式に関する。

〔従来の技術〕

初めての土地を車で訪れる場合において、迷うことなく目的地に辿り着くためには、予め道路地図等により充分な走行ルートの検討が必要である。この走行ルートの検討では、まず、走行ルートを選定しなければならないことは勿論であるが、その選定したルートを間違いなく走行するため、走行ルートにおいて右左折する交差点や特徴物等による目印、そこまでの走行距離等の道路情報を覚えておくことが必要である。しかし、道路網が錯綜してくると、初めて走行するルートでは、曲がるべき交差点の名称や目印となる特徴物を忘れてしまったり、走行しながらの確認が容易でないため、曲がるべき交差点の名称や目印となる特徴物を見落として現在位置が判らなくなったりして、車の流れにスムーズにのれないだけでなく途中で立ち往生してしまうことにもなる。

ナビゲーション装置は、上記のような心配もなく初めての目的地へ安心して車で訪れることがで

きるようにルート案内を行うものであり、近年、様々な方式のものが提案されている。それらの中には、目的地までのルートを設定してディスプレイに道路地図と設定したルートを表示するものや、そのルートを間違いなく走行できるように曲がるべき交差点に関して残距離や名称、右左折等の情報を提供し、また、走行途中でのルートが確認できるように特徴物の教示等を行うもの、さらには、表示だけでなく音声による案内を行うもの等がある。

このようなナビゲーション装置では、まず、ルートの設定が必要である。ルートを設定するには、出発地及び目的地の入力を行うことが必要であり、この入力により出発地と目的地が決まると、出発地と目的地の周囲及びその間の道路情報データからルート探索処理が行われ、複数のルートの中から最適なルートが設定される。また、本出願人が既に提案している方式では、出発地から目的地まで特定のルートを設定するのではなく、各交差点等の特定点における目的地への進行道路や進行方

向を設定するものもある。この場合には、走行距離や操舵角、通過交差点等の走行情報を収集して現在位置を認識し、その現在位置において設定されている進行道路や進行方向に関する情報を提供してルートの案内を行っている。

上記のようにナビゲーション装置では、まず、出発地から目的地へ行くためにそれぞれの位置情報を入力し、ルート探索、設定を行うことが必要である。なお、現在位置をナビゲーション装置の出発地として入力できない場合には、その近くの交差点等、特定の登録位置を出発地として入力し、その出発地を通過するときにナビゲーション開始の指示を入力することになる。その意味で本願明細書において入力される出発地とは、現在位置だけでなく、現在位置の近傍で特定の位置を入力する場合に、その位置も含めて用いる。

このような出発地や目的地等の位置入力方式についても、本出願人は既に幾つか提案をしている。例えば出発地や目的地等の対象となる登録位置を観光や駐車場、レストラン等のジャンルに分け、

また、県や都市等の地域で分けてコード化し、そのコードを入力する方式もその一つである。その場合、コード番号を直接入力する他、メニューを表示してそのメニューの中から順次選択して入力する方式も提案している。また、ノードデータを持ち、このノードデータを連結して道路網を定義し、東経、北緯の座標値により位置を入力する方式も提案している。

〔発明が解決しようとする課題〕

しかしながら、ナビゲーション装置において初期の操作として欠かせない出発地や目的地等の従来の位置入力方式は、上記のように所望の位置を入力するのに手間がかかり簡便でないという問題がある。

例えばコード入力方式では、特別のコード番号でそれぞれの位置を定義付けしているため、コードブックが必要であり、そのコードブックを逐一参照しないと位置入力ができない。しかも、コードブックは、全てのコードを搭載したものであるため膨大なものとなり、これを携帯しなければ

ならない。

また、メニュー方式では、1画面で所望の位置が入力できるということではなく、多数の画面を順次切り替えそれぞれの画面で所定項目を選択してゆかないと所望の位置が入力できない。すなわち、ナビゲーション装置に使用される表示装置は、運転席付近の比較的可見易いスペースをさいて搭載されるため、コンパクトなものが採用され、一度に表示できる情報量は少ない。したがって、画面が細かく分類されてメニュー画面数が多くなり、それだけ入力操作に時間と手間がかかるという問題がある。

座標で入力する方式も、座標テーブルがないと位置が入力できず、入力しようとする位置の座標値を座標テーブルから検索しなければならないという煩わしさがある。

本発明は、上記の課題を解決するものであって、日常的に情報を入力しやすい電話番号により簡便に目的地等の位置が入力できるナビゲーション装置の位置入力方式を提供することを目的とするも

のである。さらに他の目的は、交差点や特徴物、名称が特定されなくても電話の設置された任意の位置を目的地等として入力できるようにすることである。

〔課題を解決するための手段〕

そのために本発明は、指定された出発地と目的地を基に走行ルートを設定し、走行案内を行うナビゲーション装置において、複数の位置に関する情報からなる位置データ及び該位置データの格納情報と電話番号とを対応させるリストを有し、出発地や目的地を指定するデータとして電話番号を入力することにより位置データを読み出し可能にしたことを特徴とするものであり、さらには、リストに特定の情報を有し、入力された電話番号が登録されていない場合には特定の情報を与えるようにしたことを特徴とするものである。

〔作用及び発明の効果〕

本発明のナビゲーション装置の位置入力方式では、複数の位置に関する情報からなる位置データ及び該位置データの格納情報と電話番号とを対応

させるリストを有するので、出発地や目的地を指定するデータとして電話番号を入力すると、リストで電話番号のマッチング処理を行うことにより位置データの格納情報を得ることができ、位置データを読み出すことができる。電話番号は、案内情報や名刺、電話帳から簡便に入手でき、10桁前後の数値を入力すればよいので、出発地や目的地の位置入力が容易になる。

また、リストに特定の情報を有し、入力された電話番号が登録されていない場合には特定の情報を与えるので、リストに登録されていない電話番号が入力されても、特定の情報として電話番号の市内局番から特定される地域の代表的な位置情報を与えるようにすることができる。したがって、電話番号を入力することによって、少なくともその電話番号の近傍の位置を入力することができる。

〔実施例〕

以下、図面を参照しつつ実施例を説明する。

第1図は本発明に係るナビゲーション装置の位置入力方式の1実施例を示す図であり、1は表示

部、2は表示制御部、3は入力部、4は入力モード認識部、5は処理部、6は記憶部、7は電話番号リスト、8は位置データ、9はルート探索部、10はコードリストを示す。

第1図において、表示部1は、運転席付近の見易い位置に設けられる液晶ディスプレイやCRTディスプレイ、その他のディスプレイであり、例えばルート設定する場合に出発地や現在位置、目的地を入力するためのメニューを表示するものである。そして、ルート設定後のナビゲーション実行時にはコース案内情報等の所謂ナビゲーション情報が表示される。入力部3は、例えば表示部1の画面上に設けられ、画面のタッチ位置を検出するものであり、別にキーパネルを組み合わせたものでもよいことは勿論である。表示制御部2は、表示部1の表示内容を制御するものであり、入力モード認識部4は、入力部3でのタッチ位置は同じであっても画面によりその入力情報は異なるので、物理キーから論理キーへの変換を行い表示画面に応じて入力部3の入力情報、入力モード等を

認識するものである。記憶部6は、電話番号リスト7や位置データ8を格納したものであり、電話番号リスト7は、電話番号(TELE)毎に、位置データ8のブロック番号(ブロック№)、位置データ8の位置番号(位置№)からなる情報を有し、位置データ8は、位置番号(位置№)毎に名称や位置等の具体的な位置情報を有する。処理部5は、内部メモリを有し、適宜記憶部6からブロック単位でデータを読み込んで表示画面と入力情報、入力モードに従って入力データの処理及び表示データの処理を行うものである。本発明に係るナビゲーション装置の位置入力方式では、位置入力モードにおいて、電話番号が入力されると、電話番号リスト7を読み込んで電話番号のマッチング処理を行い、そのブロック番号(ブロック№)から位置データ8のブロックを読み込んで位置番号(位置№)のマッチングした位置情報を保持する。このように記憶部6は、必要なデータを格納する例えばCD(コンパクトディスク)等で構成し、必要に応じて処理部5の内部メモリに読み込

んで保持し、処理することにより処理速度の向上と内部メモリサイズの小容量化を図っている。ルート探索部9は、出発地又は現在位置と目的地の位置情報が保持されると、ルート探索を行い目的地までの最適ルートを設定するものである。

なお、ナビゲーション装置としての構成では、後述するようにさらに、交差点情報や道路情報、道路を定義するノード情報等からなる道路網データ、案内データ等を記憶部6に有し、また、走行中の現在位置を追跡する距離検出部や舵角検出部等が処理部5の下にあって、処理部5においてこれらの情報が処理され、ルートデータが生成されて表示部1を通してルート案内情報が出力される。ルート案内情報の出力手段としては、表示部だけでなく音声出力手段も併せて用いるようにしてもよい。

次に、具体的な処理の例を説明する。

第2図は位置入力モードでの処理の流れを説明するための図、第3図は表示画面での位置入力例を示す図、第4図は電話番号の構成例を示す図で

のモードによる位置入力処理を行う。

第3図(a)の画面の場合でのモード判断は、「ジャンル入力」キーが操作されることなくテンキーが操作されたときはYES、「ジャンル入力」キーが操作されたときはNOとなる。勿論、画面にそれぞれのモードを選択するキーを設け、必ずモード選択をするように構成する等、画面の構成や操作のモードは、種々変更可能であることはいうまでもない。

⑤ 電話番号入力処理又はその他のモードによる入力処理がなされ、位置データが読み出されると、その確定された位置情報を出発地又は目的地として登録する。

そして、上記③の電話番号入力処理は、第2図(b)に示すように、

① まず、グループ認識を行う。これは、第1図に示す記憶部6の電話番号リスト7を複数のグループに分けて構成した場合であり、「市外局番－市内局番－個別の電話番号」から予め分けられたグループ毎の認識をするものである。また、第3

ある。

ナビゲーション装置において、出発地や目的地の入力モードになると、第2図(a)に示すように、① 例えば第3図(a)に示すような位置入力モードの画面が表示される。なお、この時点では、全体の電話番号「0566-99-1111」が未入力の状態であり、「取消」は、入力ミスがあったとき等に用いるキー、「ジャンル入力」は電話番号入力「TELENO」のモードでなく、ジャンル分けされたコード入力を行うときに用いるキーである。同図(b)は電話番号を入力するのに「－」を用いず市外局番、市内局番、個別の電話番号を連続して入力する例を示したもので、この場合には、数値入力後に「終了」キーを操作する。前者の場合には、合計桁数が9桁以上で「－」の後に4桁の数値が入力されていることを条件として自動的に終了を認識すればよい。

② 入力モードが電話番号入力モード「TELE」か否かを判断し、YESの場合には次の③の電話番号入力処理を行い、NOの場合には④のその他

図(a)に示すように「－」を挿入する入力方式では、「－」の認識によって市外局番、市内局番、個別の電話番号が区分できる。しかし、第3図(b)に示すように「－」を用いずに市外局番、市内局番、電話番号を連続番号で入力する場合には、個別の電話番号の「1111」を入力し「終了」キーが操作されたことを条件として入力された番号を確定させ、市外局番、市内局番、電話番号を区分することが必要となる。特に、市外局番、市内局番は、例えば東京都区内が2桁、3桁で、都下が4桁、2桁、横浜市内が3桁、3桁のように地域によって異なる。したがって、それぞれの地域毎に区分判定が必要となる。第4図は北海道の例であるが、この第4図に示す例では、「011」の次に入力された数値が「3」であればさらに続けて「01137」までが局番とされ、それ以外であれば、「011」までが局番とされる。また、個別の電話番号の区分は、末尾から4桁の数値を抽出することによって行うことができる。

③ 認識されたグループに対応する電話番号リス

トを記録部6から処理部5に読み込む。

⑬ 読み込んだ電話番号リストで入力された電話番号とのマッチング処理を行う。

⑭ 電話番号リストのブロック8を基に記憶部6から位置データのブロックを読み込み、対応する位置8の位置情報を保持する。

以上のようにして保持された位置情報が出発地や目的地の位置情報として用いられる。そして、出発地及び目的地の位置入力が終了すると、ナビゲーション装置では通常、ルート探索の処理に移行することになる。

「本発明適用ナビゲーション装置」

次に本発明が適用されるナビゲーション装置の構成例を説明する。

第5図は現在位置算出機能を有するナビゲーション装置の1実施例システム構成を示す図である。

第5図において、入力部1は、ハードキーによる入力手段、表示画面からのタッチ入力手段等で構成するものであり、目的地入力や現在位置入力、スタート入力等に用いられる。距離センサ2は、

センサ2、舵角センサ3の検出信号を処理するものである。

ナビゲーション処理部4は、入力部1から入力されたデータや情報を解析処理する入力データ処理モジュール11、距離センサ2の検出信号を処理する距離データ処理モジュール12、舵角センサ3の検出信号を処理する舵角データ処理モジュール13を有すると共に、これらの処理部から入力されたデータや情報を基に、ナビゲーションに必要な処理を行う複数の処理モジュールを有している。

処理制御モジュール14は、入力部1から入力された指示情報に従って、ナビゲーション処理部4全体の処理制御を行うものであり、必要に応じてフラグテーブル8を参照し、経路探索モジュール15、現在位置入力モジュール16、現在位置追跡モジュール17を制御する。

経路探索モジュール15は、入力部1から目的地が入力されると、道路網データ6にアクセスしながら目的地へ案内するための最適経路を探索す

車両の走行距離を検出するものであり、舵角センサ3は、操作された車両のステアリング角を検出するものである。出力部5は、ディスプレイや音声出力装置等からなり、目的地入力や現在位置入力を行う場合に必要なメニューを表示したり、コース案内情報を出力したりするものである。道路網データ5は、交差点の情報や交差点を連結する道路、道路を構成するノード列等のデータからなるものである。ナビゲーションデータ7は、コース案内情報や設定されたコース情報その他ナビゲーションに必要なデータからなるものであり、ナビゲーション処理部4で生成されたナビゲーションデータが格納される。フラグテーブル8は、ナビゲーションを行う際に必要なフラグを登録するテーブルであり、処理状況に応じてナビゲーション処理部4からアクセスされ、適宜更新処理される。ナビゲーション処理部4は、例えばコンピュータにより構成され、道路網データ6、ナビゲーションデータ7、フラグテーブル8にアクセスして、入力部1から入力されたデータや情報、距離

のものであり、ナビゲーションの対象となる道路網データ6の全交差点に目的地への最適進行方向を設定する。そして、探索したデータは、ナビゲーションデータ7に格納される。この場合、或る交差点に着目すると、その交差点で設定された進行方向に従って道路を選択して走行し、さらに次の交差点でも、さらに次の交差点でも同様に繰り返し設定された進行方向に従って道路を選択して走行することによって、最適経路で目的地に到着できるように設定される。ここで、最短ではなく最適とした意味は、経路探索において、道路の幅や通過交差点の多寡、交通量その他の走行条件等を考慮すると、これらの重み付けによっては、必ずしも絶対距離では最短とならないが、走行時間により換算した場合には最短となるような探索も含むからである。このように交差点に注目して経路探索、ナビゲーションを行う方式において、本発明を適用する場合には、後述する経路端ノード列作成処理を行って入力位置から近くの交差点までを連結すればよい。以下、現在位置及び目的地

ではこのような処理が付加されるものとする。そこで、以下の説明では、交差点までとする。

現在位置入力モジュール16は、入力部1から現在位置が入力されると、道路網データ6及びナビゲーションデータ7から現在位置を認識してそれに対応する交差点形状や方位、目印、交差点名、進行方向道路等を描画し、出力部5の画面に表示すると共にナビゲーションに必要なデータをセットするものである。そして、スタート入力により現在位置の追跡が起動するように必要なフラグをセットする。

現在位置追跡モジュール17は、経路探索が行われてナビゲーションデータが設定され、現在位置が入力されると起動され、距離センサ2、舵角センサ3の信号とナビゲーションデータ7、道路データ6を基に交差点を検出して交差点毎に現在位置の認識処理を繰り返し行いつつ現在位置を追跡するものである。その際の状態をフラグとして適宜フラグテーブル8にセットし、また、このフラグを参照することによって各処理ステップへの

移行を判断している。そのためさらに初期位置設定18、センサ検出19、残距離計算20、屈曲点検出21、道路の選択22、距離誤差修正23等のサブモジュールを有している。

残距離計算20は、交差点により現在位置を検出するために常に交差点までの残距離を計算し、交差点までの残距離が所定の値になると交差点の認識処理を行うようにするものである。屈曲点検出21は、交差点の認識処理を行うものであり、交差点の誤差範囲において屈曲開始点、屈曲終了点を検出して交差点位置に対応する屈曲点位置を検出する。道路の選択22は、案内された進行方向道路を選択したか否かに関係なく、実際に交差点を通過して選択された道路を検出するものであり、この結果によって、その道路の終点交差点における進行方向の案内を行うようにしている。そして、距離誤差修正23は、検出された屈曲点位置、選択された進行道路を基に、距離誤差を修正しその進行道路における現在位置を求める。すなわち、ここでは、その現在位置道路における終点

交差点までの残距離が求められることになる。

このように、交差点で現在位置を検出し、その進行道路を認識することによって、その道路の終点交差点に関する案内を行ったにもかかわらず、その交差点で案内どおりに進行しなくても、その選択した道路に基づいた案内を行うことができ、ナビゲーションが続行される。つまり、交差点毎の進行方向の案内は単なる案内であって、その道路を進行方向としなくてもナビゲーションは続行が可能な構成となっている。

次に、さらに詳細に現在位置算出機能を有するナビゲーション装置による処理内容を説明する。まず、処理内容の説明に先立って、現在位置算出機能を有するナビゲーション装置で用意されるデータ構造の例を説明する。

第7図は道路網と交差点データ、道路データ及びノード列データのデータ構造例を示す図である。

いま、例えば第7図(a)に示すような交差点番号1~Ⅶ、道路番号①~⑭からなる道路網がある場合、交差点データは同図(b)、道路データは同図(c)、

ノードデータは同図(d)に示すようなデータ構造を持つ。

交差点データは、同図(b)に示すように交差点番号1~Ⅶに対応して少なくとも当該交差点が始点となっている道路のうち一番小さい道路番号、当該交差点が終点となっている道路のうち一番小さい道路番号、当該交差点の位置(東経、北緯)、交差点名の情報を持っている。

また、道路データは、同図(c)に示すように道路番号①~⑭に対応して少なくとも同じ始点を持つ道路のうち次の道路番号、同じ終点を持つ道路のうち次の道路番号、交差点番号による始点、終点、ノード列ポイント、道路長さの情報を持っている。なお、図から明らかなように同じ始点を持つ道路のうち次の道路番号、同じ終点を持つ道路のうち次の道路番号は、交差点番号による始点、終点から同じ番号を検索することによって生成することができる。また、道路長さについても次のノード列データの位置情報の積算によって求めることができる。

そして、ノード列データは、同図④に示すように道路データのノード列ポイントがポイントする先頭にノード数があり、次にその数に相当するノードについてノード位置（東経、北緯）情報を持っている。つまり、道路データ毎にノード列を構成している。図示の例は、道路番号①と②のノード列を示している。

上記のデータ構造から明らかなように道路番号の単位は複数個のノードからなる。すなわち、ノード列データは道路上の1地点に関するデータの集合であり、ノード間を接続するものをアークと呼ぶと、複数のノード列のそれぞれの間をアークで接続することによって道路が表現される。例えば道路番号①に関して見ると、道路データのノード列ポイントからノード列データのA000にアクセスすることができ、ここで道路番号①は、15個のノードからなることが認識できる。

また、例えば交差点番号Vに着目した場合、ここを始点とするコースでは、まず、交差点データの出る道路の情報から道路番号⑦、次にこの道路

番号⑦に関する道路データの「同じ始点を持つ次の道路番号」の情報から道路番号⑧が検索される。そして、道路番号⑧に関する同様の情報から道路番号⑩、続けて⑦が検索される。ここで道路番号⑦は最初の道路番号であることから周囲道路として他の道路番号のものはないとの判断ができる。これは、終点に関しても同様である。このようにして交差点データや道路データを使えば各交差点について出入りする道路番号を検索することができ、また、それぞれの交差点を結ぶ経路の距離を求めることができる。さらに、これらのデータに進入禁止や右左折禁止、道路幅のような走行条件等を付加しておくことによって、例えば後述する経路探索を極め細かに行うための情報に供することができる。

次に全体の処理の流れを第6図を参照しつつ説明する。

第6図は本発明に係る現在位置算出方式を有するナビゲーション装置による全体の処理の流れを説明するための図、第8図は経路探索出力の例を

説明するための図、第9図は案内出力の例を示す図である。

(S1) まず、目的地を入力する。目的地は、先に説明したようなメニュー画面を表示することにより、数値の部分でテンキーとして例えば電話番号「0566-99-1111」がタッチ入力される。

(S2) 次に経路探索モードになり、まず、電話番号により入力された位置から近くの交差点（目的交差点）までの経路探索を行い、次に各交差点毎に目的地（目的交差点）までの最適経路方向が設定される。例えば第7図④に示す道路網において、交差点Iが目的地であるとする、第8図④に示すように各交差点II～VIIのそれぞれに目的地への進行方向が設定される。そのデータの例を示したのが同図④である。この経路探索は、目的地に近い交差点から順次に例えば目的地への最短距離となる方向を求めることによって、各交差点の進行方向を設定する。

(S3) 出発地となる現在位置を入力する。詳細

な説明は第10図により後述するが、この処理では、「現在位置追跡失敗フラグ」、「目的地案内フラグ」等をそれぞれオフにする。

(S4) 現在位置の入力処理が行われると、その位置での進行方向の案内が可能になり、走行に従って距離センサ及び舵角センサの信号を処理して現在位置の追跡を行う。詳細な説明は第11図により後述するが、この処理では、まず、初期の段階で「交差点到着フラグ」をオフにし現在位置道路の長さや終点交差点、ノード列等の情報を認識して現在位置を追跡し、案内した交差点に到着したか否かを判断することによって、案内した交差点に到着すると「交差点到着フラグ」をオンにし、案内した交差点が検出できないと「現在位置追跡失敗フラグ」をオンにする。

(S5) 現在位置の追跡処理の後、「交差点到着フラグ」がオンかオフかを調べる。

(S6) そして、「交差点到着フラグ」がオフの場合にはさらに「現在位置追跡失敗フラグ」がオンかオフかを調べ、「現在位置追跡失敗フラグ」

がオフの場合には道路網データ中の道路を走行していると判断できるのでステップS4に戻って再度現在位置の追跡を続けて行いが、「現在位置追跡失敗フラグ」がオンの場合には道路網データ中の道路以外を走行していると判断しステップS3に戻って出発地としての現在位置入力を再度行う。

(S7)「交差点到着フラグ」がオンの場合には、続いて「目的地案内フラグ」がオンになっているか否かを調べる。「目的地案内フラグ」は、後述する処理ステップS10で現在位置道路の終点交差点が目的地交差点となったときオンにするものである。この「目的地案内フラグ」がオンになっている場合には、目的地交差点までの案内が終了したことになるので、始めの目的地入力処理に戻る。

(S8)しかし、「目的地案内フラグ」が依然としてオフのままである場合には、目的地交差点が次の交差点のさらに先にあることになるので、現在位置道路の終点交差点の進行方向データを読み取り、第9図に示すように交差点形状や交差点の

特徴、目印、交差点での進行方向等を画面に描画、交差点名や交差点までの残距離と共に表示することによって、交差点の案内を出力する。また、このとき「交差点到着フラグ」もオフにする。

(S9)そして、現在位置道路の終点交差点が目的地交差点か否かを調べる。目的地交差点でない場合にはステップS4の現在位置追跡に戻る。

(S10)現在位置道路の終点交差点が目的地交差点である場合には、「目的地案内フラグ」をオンにしステップS4の現在位置追跡に戻る。

以上が全体の処理の流れである。

次に上記ステップの主な処理ルーチンについてさらに詳細に説明する。

第10図は現在位置入力の処理ルーチンを示す図である。現在地入力S3の処理ルーチンでは、まず交差点番号が入力されると、先に説明した交差点データ、道路データ、ノード列データから交差点の方位、交差点形状、交差点名、交差点の目印等を認識して第10図(a)に示すように画面に交差点名や交差点の特徴、進行方向を描画、表示す

る(S31、S32)。そして、現在位置道路の道路番号として第8図(b)に示す進行方向データに基づいて入力された交差点番号での進行方向データを記憶し、進行方向道路を第10図(a)に示すように矢印で描画する(S33、S34)。その後、スタート入力があるまで待ち、交差点通過時に運転者がスタート入力すると、「現在位置追跡処理化フラグ」をオンにすると共に、「現在位置追跡失敗フラグ」、「目的地案内フラグ」、「交差点到着フラグ」をそれぞれオフにする(S35、S36)。

第11図は現在位置追跡の処理ルーチンの例を示す図、第12図は初期位置設定の処理ルーチンの例を示す図、第13図はセンサ検出の処理ルーチンの例を示す図、第14図は残距離計算の処理ルーチンの例を示す図、第15図は車両方位及び軌跡の計算処理を説明するための図、第16図は屈曲点検出の処理ルーチンの例を示す図、第17図は道路の選択の処理ルーチンの例を示す図、第18図は距離誤差修正の処理ルーチンの例を示す

図、第19図は距離誤差修正の処理内容を説明するための図である。

現在位置追跡の処理ルーチンは、第11図に示すように

(S411)まず初めに、「現在位置追跡初期化フラグ」がオンになっているか否かを調べる。現在位置入力が行われただけで、現在位置追跡のための初期データがセットされていない場合にはオンであるので、第12図に示す初期位置設定を行う。この設定が行われると「現在位置追跡初期化フラグ」がオフにされるので、以後は次のステップへ直接移行する。

初期位置設定では、第12図に示すように「現在位置道路の道路番号」に従って、道路長さ、終点交差点、ノード列を道路データ、ノード列データから読み込み、これらを「現在位置道路の長さ」、「現在位置道路の終点交差点」、「現在位置道路のノード列」にセットする。さらに、距離センサの信号を読み込んで、その値を「現在の距離センサ値」にセットする。そしてここで、「現



在位置追跡初期化フラグ」、「誤差範囲内フラグ」、「誤差範囲通過フラグ」、「屈曲終了待ちフラグ」、「屈曲点検出フラグ」、「屈曲検出中フラグ」等をオフにし「現在位置道路の長さ」を「交差点までの残距離」にセットする。

(S412) 次に第13図に示すセンサ検出の処理を行う。この処理では、まず、距離、舵角についてそれぞれ現在のセンサ値を前のセンサ値としてセットした後に、センサ値を読み込んで現在のセンサ値とする。そして、進んだ距離に対応した舵角の変化分を求め、現在位置での舵角  $Sta$  を求める。

(S413) ループカウンタ  $i$  を0にしてから、ループカウンタ  $i$  が進んだ距離になるまでループカウンタ  $i$  をインクリメントしながら残距離計算以下の処理を繰り返し行う。残距離計算では、第14図に示すように交差点までの残距離を逐次減算をして更新しながら交差点誤差範囲距離内に入ったか否かを判断して、交差点までの残距離が交差点誤差範囲距離内に入るまでは、「誤差範囲内

$$\begin{aligned} &= (X(i-1) + dx, Y(i-1) + dy) \\ dx &= d \times \cos(\pi - Ang(i)) \\ dy &= d \times \sin(\pi - Ang(i)) \end{aligned}$$

の式を用いて求めることができる。

また、屈曲点検出では、第16図に示すように(S421) まず、「屈曲終了待ちフラグ」、及び「屈曲検出中フラグ」がそれぞれオフで、舵角が閾値より大きいかな否かを判断し、YES(曲がり始めた)の場合には、「屈曲開始点」を現在の距離(「前の距離センサ値」+ $i$ )にセットして「屈曲検出中フラグ」をオンにする。つまり、ここで初めて舵角が閾値より大きくなるので、屈曲検出の処理状態に入る。しかし、NO(屈曲検出中、或いは舵角が閾値以下の所謂見做し直線走行中)の場合には、

(S422) 続いて「屈曲検出中フラグ」がオンで舵角が閾値より小さいかな否かを判断し、YES(前回に屈曲を検出し今回曲がり切った)の場合には、回転角度を現在の方位と屈曲開始点での方位との差にセットする。そして、回転角度が予め

フラグ」をオフにし、残距離計算を繰り返す行う。交差点までの残距離が交差点誤差範囲距離内に入ると「誤差範囲内フラグ」をオンにする。そして、「誤差範囲内フラグ」をオンにした後に交差点誤差範囲距離内から出してしまうと「誤差範囲通過フラグ」をオンにする。

(S414) 「誤差範囲内フラグ」がオンになると、車両方位及び軌跡の計算を行って第16図に示す屈曲点検出を行う。

車両方位及び軌跡の計算は、第15図に示すように一定距離  $d$  の走行毎に走行距離と舵角  $Sta$  をサンプリングし、前回の車両方位  $Ang$  と軌跡  $(X, Y)$  から新たな車両方位及び軌跡を計算してメモリに保持してゆくものである。例えば車両方位  $Ang(i)$  は、一定距離毎にセンサから読み込んだ舵角  $Sta(i)$  及び予め記憶した舵角に対する車両回転角度  $\theta(st)$  より、

$$Ang(i) = \theta(Sta(i)) + Ang(i-1)$$

また、車両軌跡  $(X(i), Y(i))$  は、

$$(X(i), Y(i))$$

決められた最小検出回転誤差より大きいかな否かを調べ、回転角度が予め決められた最小検出回転誤差以下の場合には交差点を曲がったとは判断できないので「屈曲検出中フラグ」をオフに戻すが、回転角度が予め決められた最小検出回転誤差より大きければ交差点を曲がったと判断できるので、ここで「屈曲検出中フラグ」をオフにすると共に「屈曲終了待ちフラグ」をオンにして「屈曲終了待ち位置」を現在の距離にセットする。これは、屈曲点位置を計算するのに用いられる。また、NO(屈曲終了、依然屈曲検出中、或いは直線走行中)の場合には、

(S423) さらに、「屈曲点終了待ちフラグ」がオンで、(「屈曲点終了待ち位置」+10m)が現在の距離より小さいかな否かを判断し、YES(屈曲終了から10m走行した)の場合には、回転角度を現在の方位と屈曲開始点での方位との差にセットする。そして、回転角度が予め決められた最小検出回転誤差より大きいかな否かを調べ、回転角度が予め決められた最小検出回転誤差以下の

場合には「屈曲点終了待ちフラグ」をオフにするが、回転角度が $20^\circ$ より大きければ「屈曲点検出フラグ」をオン、「屈曲終了待ちフラグ」をオフ、「屈曲終了点」を現在の距離にセットし、屈曲点位置計算を行う。

(S415)「屈曲点検出フラグ」がオンか否かを調べ、オフの場合にはカウンタ $i$ をインクリメントして同様の処理を繰り返すが、オンの場合には、道路の選択を行う。

また、交差点誤差範囲距離内に入って一旦「誤差範囲内フラグ」がオンになったが、屈曲点が検出されず「屈曲点検出フラグ」がオフのまま交差点誤差範囲距離の外に出てしまうと、先にS413で説明したように「誤差範囲内フラグ」がオフになると共に「誤差範囲通過フラグ」がオンになる。これは、交差点を曲がらずに通過した場合を含むので、この場合にも同様に次の道路の選択を行う。

道路の選択では、第17図に示すようにまず、車両の屈曲角度を計算する。そして、交差点から

の道路番号」に従って、道路長さ、終点交差点、ノード列を道路データ、ノード列データから読み込み、これらを「現在位置道路の長さ」、「現在位置道路の終点交差点」、「現在位置道路のノード列」にセットする。そして、「屈曲点検出フラグ」がオンか否かを調べ、現在の距離（「前の距離センサ値」 $+i$ ）と屈曲点位置との差 $D$ を求め、この値 $D$ を「現在位置道路の長さ」から差し引いてその結果を「交差点までの残距離」にセットする。つまり、ここでは、交差点を曲がって次の道路に進出したことにより、それまでの誤差を新たな道路に長さで修正する処理を行っている。しかし、「屈曲点検出フラグ」がオフの場合には、「交差点までの残距離」に「現在位置道路の長さ」を加えた値を新たな「交差点までの残距離」として更新する。これは、交差点を曲がらず通過した場合に相当する処理であり、例えばその交差点で「交差点までの残距離」が0になっていれば「現在位置道路の長さ」がそのまま新たな「交差点までの残距離」としてセットされることになる。

出る道路を読み込んで各連結道路の屈曲角度を計算し、連結道路屈曲角度と車両屈曲角度との差が最小の値を求める。そして、その最小値が予め決められた最大許容角度差より小さい場合には、その道路番号を「現在位置道路の道路番号」にセットし、最小値が予め決められた最大許容角度差以上の場合には、該当する道路が検出されなかったと判断し「現在位置追跡失敗フラグ」をオンにする。すなわち、全ての連結道路を対象にして車両の屈曲角度に最も近い道路を進行道路として選択するが、その間の誤差が大きい場合には、現在位置が算出できないとして現在位置の再設定を行うようにしている。

(S416)「現在位置追跡失敗フラグ」がオンか否かを調べ、オンになっている場合には第6図のステップS5、S6からS3へリターンし、オフのままである場合には、続けて距離誤差修正を行う。

距離誤差修正では、第18図に示すように、まず初期位置設定の場合と同様に、「現在位置道路

屈曲点が検出された場合と検出されない場合の誤差修正の結果を示したのが第19図である。同図(a)に示すように屈曲点位置と交差点位置との距離の差が $e d$ であるとする、同図(b)に示すように屈曲点位置を交差点位置として次の交差点までの残距離に対して $D$ の修正が行われる。

(S417)「交差点到着フラグ」をオンにし、ループカウンタ $i$ をインクリメントして同様の処理を繰り返す。

次に交差点における屈曲点の計算処理に関し説明する。

第20図は屈曲位置計算の処理ルーチンの例を示す図、第21図は車両屈曲角度計算の処理ルーチンの例を示す図、第22図は舵角と屈曲開始点、屈曲点位置、屈曲終了点を示す図、第23図は交差点から出る道路読み込み処理ルーチンの例を示す図、第24図は連結道路屈曲角度計算の処理ルーチンの例を示す図、第25図は連結道路屈曲角度の求め方を説明するための図である。

屈曲点検出処理（第16図）における屈曲点位

置計算の処理では、第20図に示すように屈曲開始点Aと屈曲終了点Cの中間距離B<sub>0</sub>、及び車両方位Angの中間方位dを求めループカウンタjを1、Bを平均距離B<sub>0</sub>にセットし、距離Bにおける車両方位Ang(B)、さらに、ループカウンタjをインクリメントしながらB<sub>0</sub>±jをBにセットした場合の車両方位Ang(B)について中間方位dと一致する点を探し、一致点Bを屈曲点位置にセットする。そして、屈曲開始点Aから屈曲点位置Bまでの距離D<sub>0..</sub>、屈曲点位置Bから屈曲終了点Cまでの距離D<sub>0...0</sub>を求める。これらの関係を示したのが第22図である。なお、屈曲開始点Aと屈曲終了点Cは、第22図(a)に示すように舵角が閾値を越えたとその点が屈曲開始点Aに設定され、舵角が閾値内に戻るとその点が屈曲終了待ち位置C'に設定され、さらにその点から10m走行した点が屈曲終了点Cに設定されている。

また、道路の選択の処理(第17図)における車両屈曲角度計算の処理では、第21図に示すよ

うに始点をもつ次の道路番号が「最初の道路番号」になるまで順次読み出すことによって、全ての道路番号を読み出し、その数を「出る道路数」にセットする。

道路の選択の処理(第17図)における連結道路屈曲角度計算の処理では、第24図に示すように、まず現在位置道路の道路番号のノード列を道路データ、ノード列データより読み込み、その終点から距離D<sub>0..</sub>の点Aの座標(x<sub>A</sub>, y<sub>A</sub>)を計算する。そして、ノード列の終点Bを座標(x<sub>B</sub>, y<sub>B</sub>)とする。次に、交差点から出る道路のノード列を道路データ、ノード列データより読み込み、その始点から距離D<sub>0...0</sub>の点Cの座標(x<sub>C</sub>, y<sub>C</sub>)を計算する。さらに、これらの座標から直線AB、BC上の点でかつこれらの直線に内接する円上の点A'、C'を、弧(A'C')の長さがD<sub>0..</sub>+D<sub>0...0</sub>となるように内接円を想定し、点A'、C'の座標を計算し、内接円と∠ABCを2等分する直線との交点を点B'として求める。この結果得られる∠A'B'C'を「連結道路屈

曲角度」にセットする。これらの関係を示したのが第25図である。

曲曲角度計算が終了すると、次に交差点から出る道路読み込みを行うが、この処理では、第23図に示すように、まず現在位置道路の終点交差点の出る道路番号を交差点データから読み込み、その道路番号をそれぞれ「道路番号」と「最初の道路番号」にセットする。そして、ループカウンタjに0をセットしてする。次に、ループカウンタjをインクリメントしながら道路データから同

様に「屈曲点検出フラグ」がオフか否かを調べ、オンの場合には、屈曲終了点、屈曲点位置、屈曲開始点がそれぞれ屈曲点検出の処理(第16図)で求められているが、オフの場合には、これらの値が求められていないので、次の値がセットされる。すなわち、屈曲終了点に現在の距離(「前の距離センサ値」+i)、屈曲点位置に現在の距離から交差点誤差範囲距離を差し引いた値、また、屈曲開始点に現在の距離から交差点誤差範囲距離の2倍を差し引いた値とする。そして、これらの3点の座標から第22図に示す∠ABCを車両屈曲角度として求める。

車両屈曲角度計算が終了すると、次に交差点から出る道路読み込みを行うが、この処理では、第23図に示すように、まず現在位置道路の終点交差点の出る道路番号を交差点データから読み込み、その道路番号をそれぞれ「道路番号」と「最初の道路番号」にセットする。そして、ループカウンタjに0をセットしてする。次に、ループカウンタjをインクリメントしながら道路データから同

様に「屈曲点検出フラグ」がオフか否かを調べ、オンの場合には、屈曲終了点、屈曲点位置、屈曲開始点がそれぞれ屈曲点検出の処理(第16図)で求められているが、オフの場合には、これらの値が求められていないので、次の値がセットされる。すなわち、屈曲終了点に現在の距離(「前の距離センサ値」+i)、屈曲点位置に現在の距離から交差点誤差範囲距離を差し引いた値、また、屈曲開始点に現在の距離から交差点誤差範囲距離の2倍を差し引いた値とする。そして、これらの3点の座標から第22図に示す∠ABCを車両屈曲角度として求める。

「交差点到着フラグ」；オンの場合には次交差点の案内出力を行うようにするものであり、現在位置入力処理時にオフにし、交差点通過時にオンにする。

「目的地案内フラグ」；交差点到着のときにオンの場合には目的地到着と判断し、メインルーチンを始め(目的地入力処理)に戻すものであり、現在位置入力処理時にオフにし、次交差点案内出力時に次交差点が目的地であればオンにする。

「現在位置追跡失敗フラグ」；オンの場合には現在位置追跡処理ではリターンし、メインルーチンでは再度現在位置入力を行うように分岐するものであり、現在位置入力処理時にオフにし、道路の選択の際、選択すべき道路がない場合にオンに

する。

「現在位置追跡初期化フラグ」；オンの場合には初期位置設定ルーチンをコールするものであり、初期位置設定時にオフにし、現在位置入力時にオンにする。

「誤差範囲内フラグ」；オンの場合には軌跡の計算、屈曲点検出等を行うようにするものであり、交差点誤差範囲内であればオン、そうでなければオフにする。

「誤差範囲通過フラグ」；オンの場合には道路の選択を行うようにするものであり、交差点誤差範囲通過すればオンにし、そうでなければオフにする。

「屈曲点検出フラグ」；オンの場合には現在位置追跡で道路の選択を行うと共に、距離誤差修正で現在位置の再設定を行い、オフの場合には車両屈曲角度計算で屈曲開始点、屈曲終了点、屈曲点位置の設定を行うものであり、初期位置設定時にオフにし、屈曲点が検出されるとオンにし、距離誤差修正後オフにする。

理を行うようにしたものである。

第27図に示す例によれば、交差点から一定の半径の円内に進入したときに交差点の誤差範囲内に入ったと認識するものである。例えば交差点座標 $(x_c, y_c)$ の誤差範囲内の距離を $r$ とすると、現在位置(車両位置)の座標 $(x, y)$ が

$$(x_c - x)^2 + (y_c - y)^2 < r^2$$

の条件を満足する位置に達すると、第26図(a)に示す円内に進入したとする。そして、同図(b)に示すように屈曲検出を行って、屈曲開始位置と屈曲終了位置を検出し、これらの位置の方位をそれぞれ $-20^\circ$ と $-100^\circ$ とすると、

軌跡屈曲値度

$$\begin{aligned} &= (\text{屈曲終了位置方位}) - (\text{屈曲開始位置方位}) \\ &= -100^\circ - (-20^\circ) = -80^\circ \end{aligned}$$

を求め、さらに屈曲点位置を検出し、

$$D_{pp} = |\text{屈曲点位置} - \text{屈曲開始位置}|$$

$$D_{pps} = |\text{屈曲終了位置} - \text{屈曲点位置}|$$

とする。

「屈曲検出中フラグ」及び「屈曲終了待ちフラグ」；屈曲検出及び屈曲終了点の検出を行うものであり、初期位置設定でオフにする。そして「屈曲検出中フラグ」は舵角が閾値を越えるとオンにし回転角度が最小検出回転角度を越えたか否かによりこれらのオン/オフを制御する。

次に本発明の他の実施例を説明する。

第26図は現在位置算出機能を有するナビゲーション装置の他の実施例で現在位置追跡の処理ルーチンの例を示す図、第27図は屈曲点検出を説明するための図である。

上記の実施例は、道路番号とその道路番号の終点交差点からの残距離により現在位置を追跡するものであるが、第26図に示す実施例は、交差点を座標により認識し現在位置を追跡するものである。従って、現在位置追跡の処理において、屈曲点検出までは、上記実施例と同じであるが、上記実施例で行った道路の選択から距離誤差修正までの処理に代えて、交差点通過時の処理を交差点進入方位、屈曲角度比較の処理、現在位置修正の処

第28図は交差点進入方位、屈曲角度比較処理を説明するための図、第29図は第28図の処理に対応する処理ルーチンの例を示す図、第30図は交差点に入る道路読み込みの処理ルーチンの例を示す図、第31図は現在位置修正処理をせつめするための図、第32図は第31図の処理に対応する処理ルーチンの例を示す図である。

交差点進入方位、屈曲角度比較処理では、第28図(a)に示すように、交差点番号IVに入る道路を交差点データの入る道路及び道路データの同じ終点をもつ道路より読み出し、これらの道路より進入した場合の方位を道路に対するノード列上の交叉点位置から $D_{ps}$ の位置の方位として求める。その方位が、例えば同図(b)に示すようにそれぞれ

道路番号	交差点IVへの進入方位
③	150°
⑥	-100°
⑦	60°
⑩	-25°

であるとする、第27図(b)に示す例では、屈曲開始位置方位(−20°)に最も近い進入方位をもつ道路番号⑩を進入道路とする。

進入道路が求まると、次に交差点番号Ⅳから出る道路および道路データから同じ始点をもつ道路を読み出し、これらの道路へ進入道路⑩から進行情況の場合の方位変化を求める。この処理では、出る道路のノード列上の交差点位置からD...の位置での方位と進入道路における屈曲開始位置方位との変化として求める。従って、進入道路⑩と各出る道路④、⑤、⑧、⑨との関係を見たとき、

道路番号	屈曲終了位置方位及び屈曲角度
④	$-30^{\circ} - (-25^{\circ}) = -5^{\circ}$
⑤	$80 - (-25^{\circ}) = 105^{\circ}$
⑧	$-120 - (-25^{\circ}) = -95^{\circ}$
⑨	$155 - (-25^{\circ}) = 180^{\circ}$

となる。従って、第27図(b)に示す例では、軌跡屈曲角度−80°であるので、これに最も近い屈曲角度の道路⑩→⑨が進行方向として判定される。

までインクリメントしつつ第28図(b)に示す進入道路を判定を行い、続いて、ループカウンタkを0にセットし、これを出る道路数までインクリメントしつつ屈曲角度を計算して第28図(c)に示す進行方向道路の判定を行う。そして、ここで求めた進行方向道路を現在位置道路の道路番号としてセットする。

現在位置修正は、上記現在位置道路の道路番号を求めた後行われるが、この処理では、第31図(a)に示すように現在位置座標を道路⑩のノード列上の屈曲終了座標とし、現在位置方位をノード列上の屈曲終了位置方位(−120°)とする。そして、同図(b)に示すように次の交差点として道路番号⑧の終点交差点Ⅴを道路データより読み出し、この交差点座標を交差点データより読み出して、この交差点Ⅴの誤差範囲内に車両が進入するのを監視する。

以上の現在位置修正の処理ルーチンを示したのが第32図である。第32図に示す処理ルーチンでは、まず、

以上の交差点進入方位、屈曲角度比較の処理ルーチンを示したのが第29図である。第29図に示す処理ルーチンでは、まず、

交差点に入る道路読み込みの処理を行う。

この処理では、第30図に示すように現在位置道路の終点交差点の入る道路番号を交差点データから読み込み、この番号をそれぞれ「道路番号」、「最初の道路番号」にセットする。そして、ループカウンタjに0をセットしてする。次に、ループカウンタjをインクリメントしながら道路データから同じ終点をもつ次の道路番号が「最初の道路番号」になるまで順次読み出すことによって、全ての道路番号を読み出し、その数を「入る道路数」にセットする。つまり、先に第23図で説明した交差点から出る道路番号読み込みの処理に対応する処理を行う。

同様に、第23図で説明した交差点から出る道路番号読み込みの処理を行い、ループカウンタjを0にセットする。

そして、まず、ループカウンタjを入る道路数

現在位置道路の道路番号の終点交差点、ノード列を道路データより読み込み、これらをそれぞれ「現在位置道路の終点交差点」、「ノード列」とする。

また、「現在位置道路の終点交差点」の東経、北緯の座標を交差点データより読み込み、「次交差点東経、北緯の座標」とする。

そして、「ノード列」の始点からD...の距離のノード列上の座標を現在位置の座標(X〔「前の距離センサ値」+i〕、Y〔「前の距離センサ値」+i〕)とし、「ノード列」の始点からD...の距離でのノード間の方位を現在位置の方位とする。

「経路端ノード列作成及び経路探索」

次に、電話番号により入力した位置から近くの交差点までのノード列を作成する経路端ノード列作成処理及び交差点からの経路探索処理について説明する。なお、上記の経路探索では、目的地を入力して目的地への進行方向を各交差点に設定する例を説明したが、以下の例では、出発地から目

的地までの最適経路を探索する例を説明する。

第33図は目的地の設定とそのデータ構成例を示す図、第34図は交差点列およびノード列データの構成例を示す図である。

この経路端ノード列作成処理は、電話番号入力によって位置データから得られた出発地、目的地からそれぞれ最寄り交差点までのノード列を作成するものであり、最寄り交差点から出発地又は目的地の位置情報に最も近いノードに至るまで順次ノード探索を行ってノード列を作成する。目的地を交差点と交差点との間に設定する場合の例を示したのが第3図である。例えば、第3図(a)に示すように目的地Aを交差点ⅠとⅡとの間で道路とは直接関連させない座標により位置データから与えられた場合、その目的地データは同図(b)に示すように連絡交差点Ⅰ、Ⅱで、座標が東経135°、北緯35°と表現される。また、同図(c)に示すように目的地Aを交差点ⅠとⅡとの間で一方の交差点Ⅰからの距離により与えられた場合、その目的地データは同図(d)に示すように連絡交差点Ⅰ、Ⅱ

で、連絡交差点Ⅰからの距離50mと表現される。従って、経路端ノード列作成処理では、連絡交差点からこの目的地Aのノード或いは最も近いノードまでのノード列を作成する。

経路探索処理では、右左折禁止等の進入禁止道路を除き交差点から周囲道路を検索する周囲道路検索サブルーチン、道路幅の広狭、案内の要否その他最適経路を演算するのに必要な条件を設定する最適経路条件設定サブルーチン、経路探索の終了を判定する終了条件サブルーチンを有し、指定された出発地から目的地までの最適経路を最寄りの交差点間で探索する。そして、経路探索処理により最適経路が探索されると、その経路に沿って出発地から目的地までの最寄り交差点間の交差点列及びノード列データが作成される。このデータに経路端ノード列作成処理により作成されたノード列を両端に連結したのが交差点列及びノード列データであり、そのデータ構成例を示したのが第34図である。

第34図に示すような目的地の人力設定に対す

る経路端ノード列作成処理の流れを第35図を参照して説明する。

まず、目的地の人力設定が第33図(b)に示すような座標指定であるか否かを調べ、YESの場合には次の座標指定の処理を行い、NOの場合には第33図(d)に示すような距離指定の人力設定であるので後述する距離指定の処理を行う。

(1) 座標指定の処理：第35図(a)

- ① 連絡交差点ⅠのノードをNとする。
- ② 目的地のノードをN。とすると、連絡交差点Ⅰと目的地との間の直線距離 $D = NN。$ を計算し、距離Dをメモリに記憶する。
- ③ 距離Dをメモリに記憶する。
- ④ ノードNが連絡交差点Ⅱか否かを調べる。

YESの場合には目的地が連絡交差点Ⅱであったということなので、そのノードNを記憶して終了とする。この判定は最初の処理ではありえず連絡交差点Ⅱまでのノードについて④～⑥の処理を繰り返して終了条件に達しなかったときである。NOの場合には次の処理⑤を行う。

⑤ ノードNの次のノードをN' とする。

⑥ 分割定数tを0とする。

⑦ ノードNの座標 $N。$ 、 $N。$ について

$$N。 = (1-t) N。 + t N。'$$

$$N。 = (1-t) N。 + t N。'$$

とする。

⑧ 距離Dが処理⑦により計算した座標を用いた距離 $NN。$ 、すなわちノードNと次のノードN' との間を分割定数tにより分割した点から目的地のノードN。までの距離より大きいかなを調べる。

YESの場合には次の処理⑨を行う。この場合には、前回の点より今回の点が目的地に近づいていることを示すので、さらに点をシフトさせる。NOの場合には座標 $N。$ 、 $N。$ をメモリに記憶して終了とする。すなわち、NOの場合には、前回の点までは目的地に近づいてきたが、今回の点は目的地から前回より遠くなったことを示すので、このときのノードNと次のノードN' との間が目的地に最も近いことになり、ここから連絡交差点

までのノード列を作成すればよい。

⑨ 距離DをNN<sub>0</sub>に置き換える。

⑩ 分割定数tを0.1加算する。すなわち、分割点を10%ずつシフトする。

⑪ 分割定数tが1を越えたか否かを調べる。

YESの場合には次の処理⑫を行ってノードNを次のノードN'に置き換えて処理⑨に戻る。すなわち、YESはノードNから次のノードN'へ0.1ずつ分割点がノードN'に到達したことを意味するので、処理⑫でノードを1つシフトするものである。また、NOの場合には処理⑦に戻る。すなわち、この場合には分割点がノードNと次のノードN'との間にあるので、処理⑦以降の処理を繰り返し行うものである。

(ロ) 距離指定の処理：第35図(ロ)

① 連絡交差点IのノードをNとする。

② 距離Dを0とする。

③ 距離Dをメモリに記憶する。

④ メモリに記憶したノードNが連絡交差点IIか否かを調べる。

差点までのノード列を作成すればよい。NOの場合には次の処理⑤を行う。この場合には、前回の点より今回の点が目的地に近づいていることを示すので、さらに点をシフトさせる。

⑥ 分割定数tを0.1加算する。すなわち、分割点を10%ずつシフトする。

⑦ 分割定数tが1を越えたか否かを調べる。

YESの場合には次の処理⑧を行ってノードNを次のノードN'に置き換え、さらに距離Dに距離NN'を加算した距離を新たな距離Dとして置き換えて処理⑥に戻る。すなわち、YESはノードNから次のノードN'へ0.1ずつ分割点がノードN'に到達したことを意味するので、処理⑧でノードを1つシフトするものである。また、NOの場合には処理⑦に戻る。すなわち、この場合には分割点がノードNと次のノードN'との間にあるので、処理⑦以降の処理を繰り返し行うものである。

次に上記のネットワークデータにより経路探索する場合の処理の流れを第36図により説明する。

YESの場合には連絡交差点IIのノードNを記憶して終了とする。NOの場合には次の処理⑤を行う。

⑤ ノードNの次のノードをN'とする。

⑥ 分割位置定数tを0とする。

⑦ ノードNと次のノードをN'との間を分割位置定数tによる分割点の座標N<sub>0</sub>'、N<sub>1</sub>'

$$N_{0}' = (1-t)N_0 + tN_1'$$

$$N_{1}' = (1-t)N_1 + tN_2'$$

を求める。

⑧ 距離DにノードNから処理⑦により計算した座標N<sub>0</sub>'、N<sub>1</sub>'の点N'までの距離NN'を加算した距離が入力された目的地から連絡交差点Iまでの距離より大きいかなを調べる。

YESの場合には座標N<sub>0</sub>'、N<sub>1</sub>'をメモリに記憶して終了とする。すなわち、YESの場合には、前回の点までは目的地に近づいてきたが、今回の点は目的地から前回より遠くなったことを示すので、このときのノードNと次のノードN'との間が目的地に最も近いことになり、ここから連絡交

ここでL(c)は距離、F(c)はフラグ、R(c)は通過してきた道路番号、s<sub>0</sub>、s<sub>1</sub>は出発地の両隣の交差点番号、e<sub>0</sub>、e<sub>1</sub>は目的地の両隣の交差点番号である。また、cは交差点番号、フラグF(c)は「0」が未探索、「1」が探索中、「2」が探索終了を示す。

① 全ての交差点について

距離L(c)に無限大(∞)

フラグF(c)に「0」(未探索)

にセットする。この初期設定によりまず全ての交差点が未探索となり、出発地からの距離が無限大となる。

② 出発地の両隣の交差点番号s<sub>0</sub>、s<sub>1</sub>に対応する距離L(s<sub>0</sub>)、L(s<sub>1</sub>)に出発地からの距離を入れ、さらに出発地の両隣の交差点番号s<sub>0</sub>、s<sub>1</sub>に対応するフラグF(s<sub>0</sub>)、F(s<sub>1</sub>)にそれぞれ「1」、通過してきた道路番号R(c)に出発地からの道路番号をセットする。

③ フラグFが「2」でなく且つ距離L(c)が最

小となる交差点番号  $c_0$  を検索する。

- ④ 周囲道路検索サブルーチンを実行し、交差点番号  $c_0$  を始点とする周囲道路を検索する。

- ⑤ 周囲道路があるか否かを調べる。

YES の場合には次の処理⑥に移り、NO の場合には処理⑪に移る。

- ⑥ 最適経路条件設定サブルーチンを実行し、最適経路を探索するための道路状況その他の条件を設定する。

- ⑦ その道路の終点の交差点番号を  $c_1$ 、道路の長さを  $l$  とする。

- ⑧ その道路の終点の交差点までの距離  $P$  を計算する。

$P = L(c_0) + l$  を計算する。

ここで  $L(c_0)$  は出発地から交差点番号  $c_0$  までの距離であり、 $P$  は交差点番号  $c_0$  からその道路（探索中の道路）を通して終点の交差点番号  $c_1$  までの距離となる。

- ⑨  $P < L(c_1)$  で且つ  $F(c_1) \neq 2$  か否かを調べる。

には処理⑩に移る。

- ⑩ 交差点データから現在いる交差点  $c_0$  が始点となっている道路番号を取り出し記憶する。

- ⑪ 道路データを参照し探索中の当該交差点  $c_0$  にくる道路番号における禁止道路を取り出す。

- ⑫ 今取り出した道路が禁止道路か否かを調べる。

YES の場合には処理⑬に移り、NO の場合には次の処理⑭に移る。

- ⑭ 今取り出した道路を周囲道路として記憶し、リターンする（第36図(a)の処理⑮へ移る）。

- ⑮ 道路データから前に探索した道路と同じ始点を持ち、番号が次の道路番号を取り出す。

- ⑯ 最初探索した道路と今取り出した道路が同じか否かを調べる。

YES の場合には次の処理⑰に移り、NO の場合には処理⑱に戻る。

- ⑰ 周囲道路なしと判定しリターンする。

また、上記第36図に示す処理⑮の最適経路条件設定サブルーチンは、第38図に示すような処理を行うものである。すなわち、

YES の場合には次の処理⑲に移り、NO の場合には処理⑱に戻る。

- ⑲ 出発地から探索中の交差点番号  $c_1$  までの距離  $L(c_1)$  を  $P$ 、その交差点番号  $c_1$  のフラグ  $F(c_1)$  を「1」、交差点番号  $c_1$  に至るまでに通過してきた道路番号  $R(c_1)$  をその探索中の道路番号とする。

- ⑳ 処理⑲においてNOの場合には  $F(c_0)$  を「2」にセットする。

- ㉑ 終了条件確認サブルーチンを実行する。

- ㉒ 処理終了か否かを調べ、NOの場合には処理⑲に戻る、YESの場合には処理を終了とする。

以上の処理を行うことによりそれぞれの交差点番号に対応して出発地から当該交差点番号に至る最適コースの道路番号がそれぞれ交差点番号毎に設定される。

上記処理④の周囲道路検索サブルーチンは、第37図に示す処理を行うものである。すなわち、

- ① 周囲道路の検索が1回目か否かを調べる。

YES の場合には処理②に移り、NO の場合

- ① 道路データから周囲道路の大きさ  $W$  と長さ  $l$  を読み込む。

- ② 周囲道路の大きさ  $W$  が1以下であるか否かを調べる。

YES の場合には次の処理③に移り、NO の場合には処理④に移る。

- ③ 長さ  $l$  を  $a$  倍した長さを  $l$  とする。すなわち、 $D$  が1より大きい道路を通常の広い道路とし、1以下の道路を細い道路とすると、細い道路は通常の道路に対して  $a$  倍の距離の評価としている。従って、 $a$  は1より大きい数である。

- ④ 道路データから現在探索中の交差点へ通過してきた道路の案内不要データを読み込む。

- ⑤ 案内不要データと一致する周囲道路があるか否かを調べる。

YES の場合にはリターンし、NO の場合には次の処理⑥に移る。

- ⑥ さらに長さ  $l$  に  $bm$  を加算した値を新たな長さ  $l$  としリターンする。すなわち、案内不要の交差点に対して、右左折等の案内を要する交



点は、距離に換算してb m加算した評価として  
いる。

終了条件確認サブルーチンでは、第39図に示  
すように探索対象の交差点番号c。と目的等の両  
隣りの交差点番号との一致を調べ、一致したこと  
を条件に例えば終了フラグを設定する。

上記のように本発明の経路探索では、周囲道路  
の大きさや道路の案内要/不要等の走行条件を考  
慮して交差点間の距離に重み付けを行い、最短経  
路を探索する。その結果、各交差点に対応して最  
適コースに沿った道路番号情報が得られる。従っ  
て、この探索結果に従って第40図に示す処理フ  
ローに従って交差点列及びノード列のデータを作  
成することができる。

- ① 探索が終了した交差点番号をメモリに記憶す  
る。
- ② その交差点にきた道路番号の始点をメモリに  
記憶する。
- ③ その交差点が出発地の両隣りの交差点か否か  
を調べる。

み出して出力すればよい。

上記のように経路探索時は、右左折禁止データ  
をチェックしながら探索し、右左折禁止が入らな  
いコースを探索する。

なお、上記の実施例に限定されるものではなく、  
種々の変形が可能である。例えば上記の実施例で  
は、出発地から目的地までナビゲーションのコース  
をノード列と交差点列からなるデータ列により  
作成したが、ナビゲーションのためのデータ列と  
しては、ノード列と交差点列のいずれか一方のみ  
でもよい。交差点列のみの場合には、交差点列の  
それぞれのデータにナビゲーションのための写真  
や案内情報その他必要な情報をもたせるようにす  
ることは勿論である。また、交差点列のデータを用  
いてナビゲーションを実行する場合には、例えば  
単に押しボタンスイッチを設け、運転者が交差  
点を通過する毎に押しボタンスイッチを操作する  
と、次の交差点データを読み出すようにすること  
によって、簡単な操作によるナビゲーションを実  
行させることができる。

YESの場合には次の処理④に移り、NOの  
場合には処理②に戻る。

- ④ 記憶した交差点番号列の前と後に出発地番号、  
目的地番号を加えて交差点列とする。
- ⑤ 道路データを参照して交差点間のノード列を  
取り出し、ノード列をつくる。
- ⑥ 案内不要データを使い交差点列から案内不要  
となる交差点を除く。

このようにして経路探索の結果から生成される  
交差点列及びノード列データの例を示したのが第  
34図である。例えば交差点列データは、第34  
図(a)に示すように交差点名、交差点番号、その交  
差点の特徴風景等を撮影した写真番号、曲がる角  
度、距離等の情報からなり、また、ノード列デー  
タは、同図(b)に示すようにそのノード位置を表す  
東経、北緯、そして交差点番号、属性、角度、距  
離等の情報からなる。しかも、これらのデータは、  
案内不要の交差点を除いた、案内を要する交差  
点のみのデータからなる。従って、ナビゲーション  
では、所定の位置に対応してこのデータを順次読

上記の実施例では、経路探索を出発地からスタ  
ートさせたが、目的地から経路探索をスタートさ  
せるようにしてもよい。また、出発地から経路探  
索をスタートして目的地に達したところで処理終  
了にしたが、全てのフラグF(c)が2になるま  
で、すなわち、全ての交差点について経路探索を  
行うようにしてもよい。特に目的地からこの経路  
探索を行うと、上記の第5図で示した例のように  
全ての交差点から目的地までの最適コース情報が  
作成されることになるので、途中でコースから外  
れた場合にも、経路探索を再度行うことなく、最  
密りの交差点から交差点列及びノード列を作成す  
ることができる。

なお、本発明は、上記の実施例に限定されるも  
のではなく、種々の変形が可能である。例えば上  
記の実施例では、電話番号リスト7を、TEL皿、  
ブロック皿、位置皿からなる構成としたが、位置  
データに関する情報としてはポイントを用いても  
よいし、TEL皿を設けず電話番号をアドレスと  
してもよい。この場合、全ての電話番号を登録し、

位置データに実情報を持たないものについては、局番で限定される地域の代表位置の情報、その他特定の情報を持つようにしてもよいし、その地域の地図情報を持ち地図を画面に表示してもよい。このようにすると、その地図上で座標により位置が入力できる。また、電話番号リストにない電話番号が入力された場合にも同様に代表位置の情報や地図情報を用いるようにしてもよい。

出発地や現在位置、目的地となる位置には、公園や観光地のように必ず電話があるとは限らない。したがって、電話番号のない場合や電話番号が判らない場合には、電話番号以外の入力により位置入力ができるように他の入力方式と併用してもよいことは勿論であり、例えば先に述べたようなコード入力方式やメニュー入力方式の他、数値入力や表示された地図上でのタッチ入力による座標入力方式、音声入力方式、カード入力方式等と併用することができる。例えばコード入力方式と併用する場合、第1図に示すようにコードリスト10を別に持つようにしてもよいし、電話番号リスト

と併合してもよい。そして、前者のコードリスト10を別に持つ場合には、電話番号リストのブロック、位置座標をコードリスト10のコード座標に代えてもよいし、後者の併合する場合には、特にモード切り替えキーを設けず、入力した番号の頭が「0」の場合には電話番号入力にし、「0」以外の場合にはコード入力にしてもよい。

また、電話番号を音声で入力するようにしてもよい。特に音声入力の場合には、認識精度が問題となり音声登録を行うものもあるが、電話番号の場合には、「0～9」までの音声を登録すればよいので、音声登録が必要な場合でも、登録音声が少ない誤認識率を少なくすることができる。

さらには、電話番号入力において、駅その他の代表的な施設の電話番号や高速道路のインターチェンジ、国道沿いのガソリンスタンド等、特に目標としやすい施設の電話番号を特定の短縮番号として設定しておき、その短縮番号で入力できるようにしてもよいし、この場合に、その位置を目的とするだけでなく、その付近の地図を表示するこ

とにより地図上で目的地を入力できるようにしてもよい。このようにすることによって、電話番号は判らないが近くに代表的な施設がある場合の目的地入力を簡便にすることができる。

以上の説明から明らかなように、本発明によれば、電話番号により位置を入力できるので、出発地や目的地を名称のある地点や交差点等のような特定の位置だけでなく、電話のある位置を自由に設定できる。

したがって、例えば走行中に迷ってしまい全く位置が判らなくなっても、近くのカソリンスタンドや店、看板等に表示された電話番号を入力することにより容易に現在位置をナビゲーション装置に入力することができ、そこを出発地として走行ルートを設定しなおし、ナビゲーション装置による新たなルート案内を受けることができる。

また、位置を入力するのに、従来の方式のようにコードブックや座標テーブル等を参照したり、複数のメニュー画面を切り替えて入力することなく、1画面で10桁前後の数値を入力するだけで

よいので、短時間に簡単に入力操作を終了させることができる。特に、コード入力や座標入力は、入力しようとする位置と直接結び付けられないが、電話番号の場合には、その位置特有の情報として日常的にも意識、記録されるので、位置との結び付きが深く、有用且つ便利な情報として用いることができる点でメリットは大きい。

#### 4. 図面の簡単な説明

第1図は本発明に係るナビゲーション装置の位置入力方式の1実施例を示す図、第2図は位置入力モードでの処理の流れを説明するための図、第3図は表示画面での位置入力例を示す図、第4図は電話番号の構成例を示す図、第5図は本発明が適用される現在位置算出機能を有するナビゲーション装置の1実施例システム構成を示す図、第6図は現在位置算出機能を有するナビゲーション装置による全体の処理の流れを説明するための図、第7図は道路網と交差点データ、道路データ及びノード列データのデータ構造例を示す図、第8図は経路探索出力の例を説明するための図、第9図

は案内出力の例を示す図、第10図は現在地入力  
の処理ルーチンを示す図、第11図は現在位置追  
跡の処理ルーチンの例を示す図、第12図は初期  
位置設定の処理ルーチンの例を示す図、第13図  
はセンサ検出の処理ルーチンの例を示す図、第1  
4図は残距離計算の処理ルーチンの例を示す図、  
第15図は車両方位及び軌跡の計算処理を説明す  
るための図、第16図は屈曲点検出の処理ルーチ  
ンの例を示す図、第17図は道路の選択の処理ル  
ーチンの例を示す図、第18図は距離誤差修正の  
処理ルーチンの例を示す図、第19図は距離誤差  
修正の処理内容を説明するための図、第20図は  
屈曲位置計算の処理ルーチンの例を示す図、第2  
1図は車両屈曲角度計算の処理ルーチンの例を示  
す図、第22図は舵角と屈曲開始点、屈曲点位置、  
屈曲終了点を示す図、第23図は交差点から出る  
道路読み込み処理ルーチンの例を示す図、第24  
図は連結道路屈曲角度計算の処理ルーチンの例を  
示す図、第25図は連結道路屈曲角度の求め方を  
説明するための図、第26図は本発明に係る現在

位置算出方式を有するナビゲーション装置の他の  
実施例で現在位置追跡の処理ルーチンの例を示す  
図、第27図は屈曲点検出を説明するための図、  
第28図は交差点進入方位、屈曲角度比較処理を  
説明するための図、第29図は第28図の処理に  
対応する処理ルーチンの例を示す図、第30図は  
交差点に入る道路読み込みの処理ルーチンの例を  
示す図、第31図は現在位置修正処理をせつめす  
るための図、第32図は第31図の処理に対応す  
る処理ルーチンの例を示す図、第33図は目的地  
の設定とそのデータ構成例を示す図、第34図は  
交差点列及びノード列データのデータ構成例を示  
す図、第35図は経路端ノード列作成処理の流れ  
を説明するための図、第36図は経路探索処理の  
流れを説明するための図、第37図は周囲道路検  
索サブルーチンの処理の流れを説明するための図、  
第38図は最適経路条件設定サブルーチンの処理  
の流れを説明するための図、第39図は終了条件  
確認サブルーチンの処理の流れを説明するための  
図、第40図は交差点列及びノード列取り出し処

理の流れを説明するための図である。

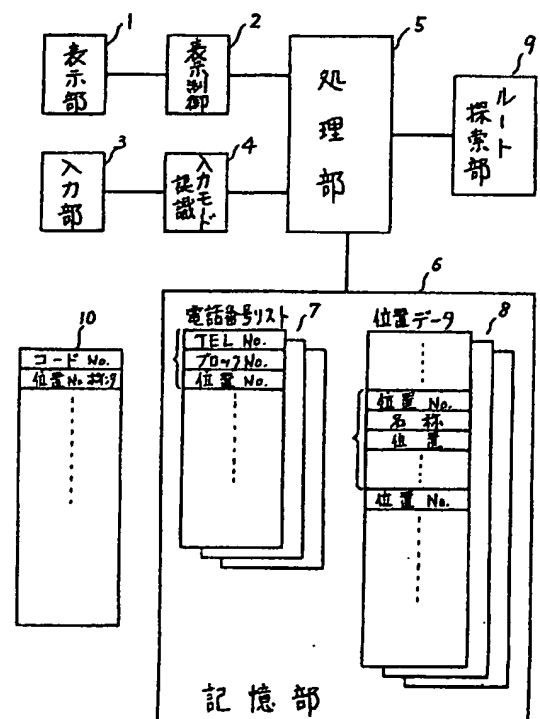
1…表示部、2…表示制御部、3…入力部、4  
…入力モード認識部、5…処理部、6…記憶部、  
7…電話番号リスト、8…位置データ、9…ルー  
ト探索部、10…コードリスト。

出 願 人 アイシン・エイ・ダブリュ株式会社

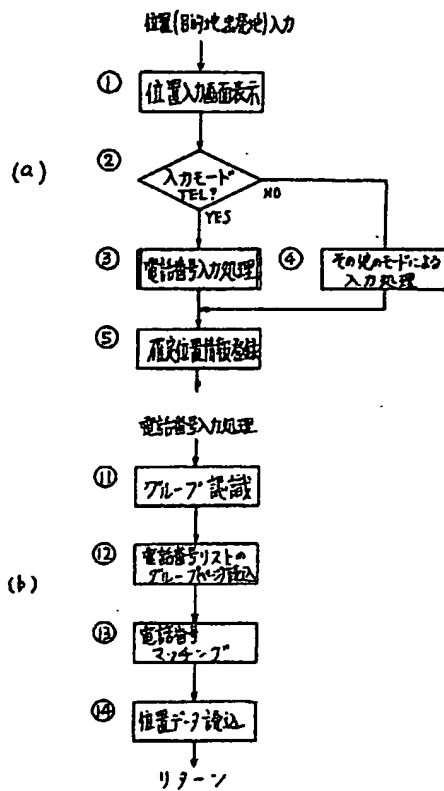
(外1名)

代 理 人 弁理士 阿 部 龍 吉 (外5名)

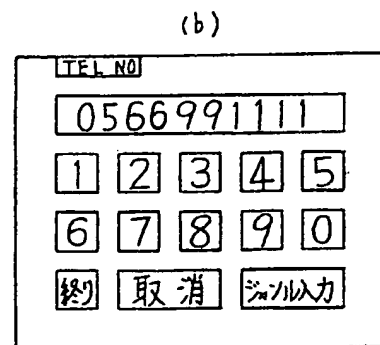
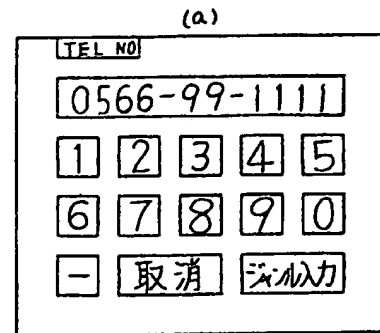
第1図



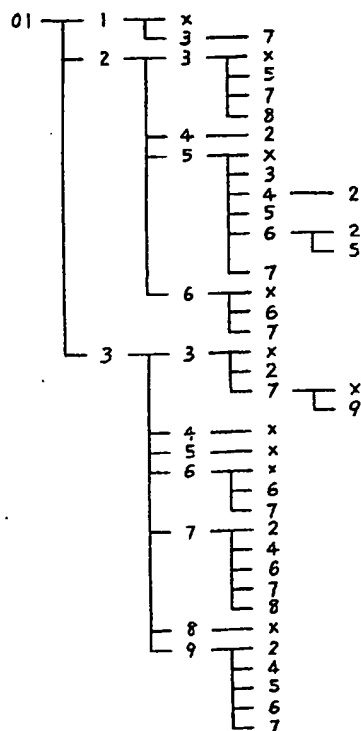
第 2 図



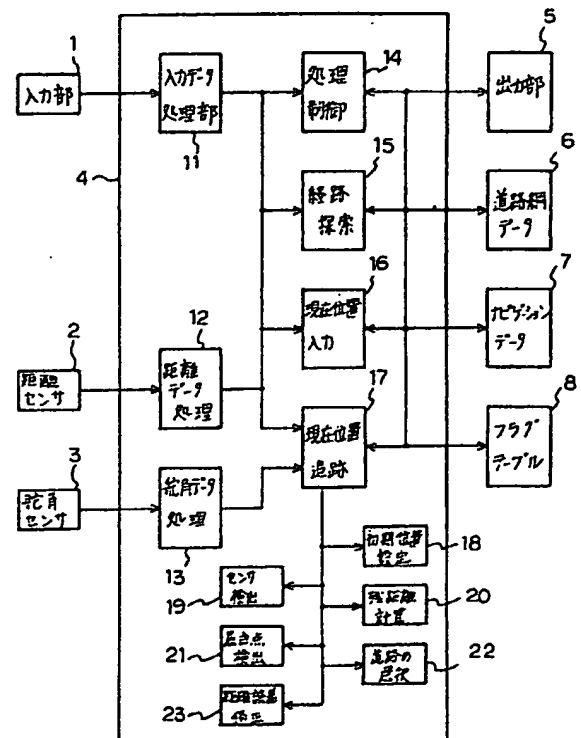
第 3 図



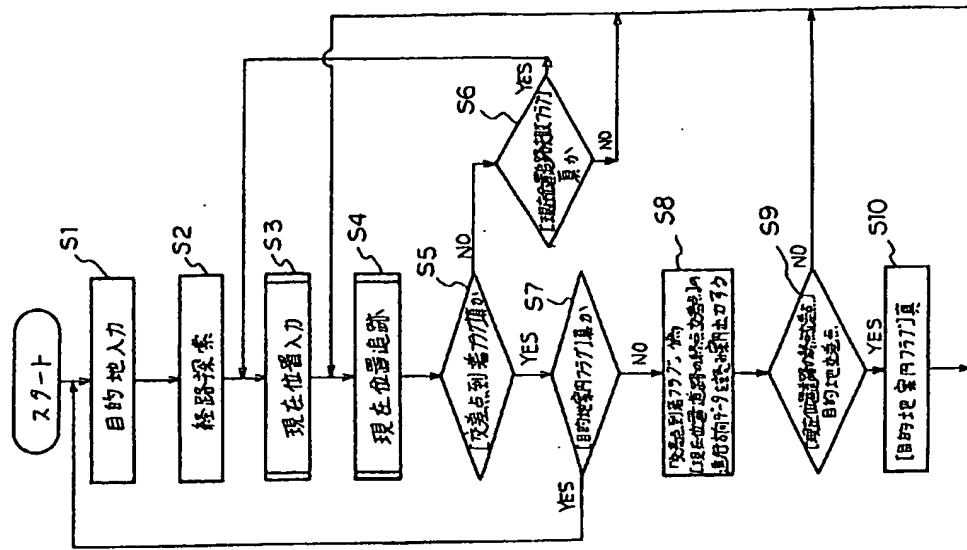
第 4 図



第 5 図

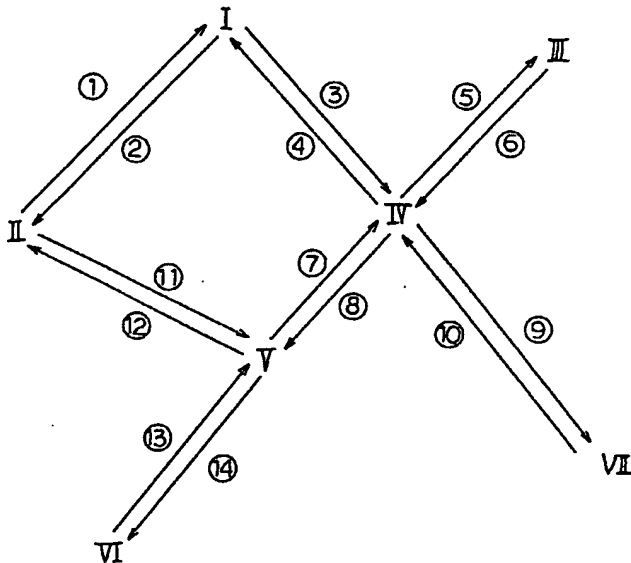


第 6 図



第 7 図 (d)

第 7 図 (a)

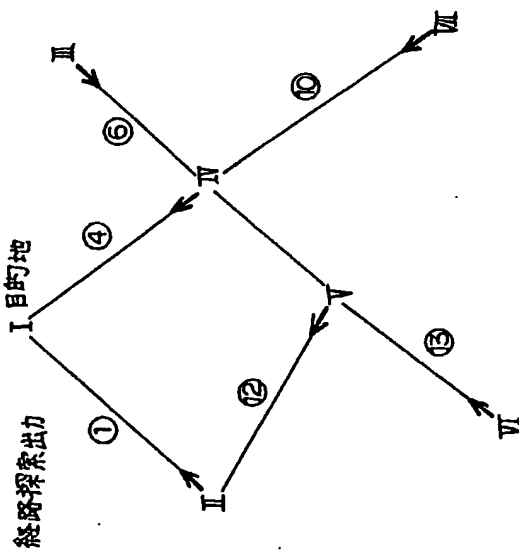


1-ド列デ-9

A000	15	← 1-ド数 (東経, 北緯)
	(10, 100)	
	(20, 110)	
	(30, 130)	
	...	
A0A0	20	
	(50, 150)	
	(40, 130)	
	...	

第8図

(a)



(b)

進行方向データ

交差点番号	進行方向
I	0
I	1
II	6
IV	4
V	12
VI	13
VII	10

第7図 (b)

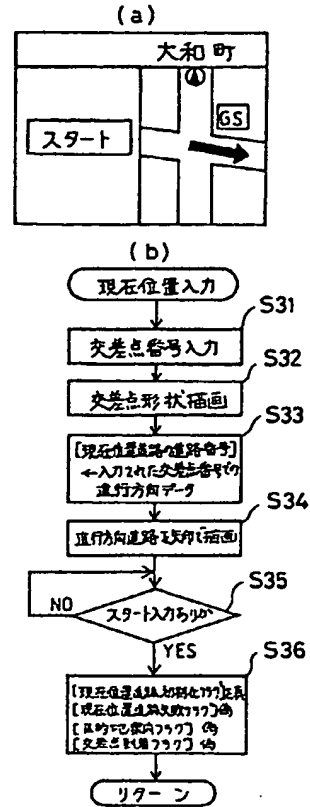
交差点データ

交差点番号	出入道路	座標(北緯)	交差点名
I	1	(50, 150)	南安城駅前
II	2	(10, 100)	南安城駅前
III	3	(150, 150)	朝日町
IV	4	(100, 100)	相生町
V	5	(50, 50)	相生町
VI	6	(10, 10)	相生町
VII	7	(150, 20)	相生町

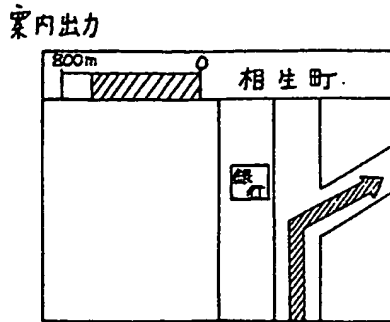
第7図 (c)

道路番号	道路名称	始点	終点	片側幅員	道路長さ
1	1	II	I	A000	1000
2	2	I	II	A0A0	1000
3	3	I	IV	A0B3	2000
4	4	IV	I	A0C0	2000
5	5	IV	III	A0DE	1500
6	6	III	IV	A101	1500
7	7	V	IV	A201	800
8	8	IV	V	A221	800
9	9	IV	VII	A253	
10	10	VII	IV	A260	
11	11	I	V	A265	
12	12	V	II	A28B	
13	13	VI	V	A2A0	
14	14	V	VI	A2B0	

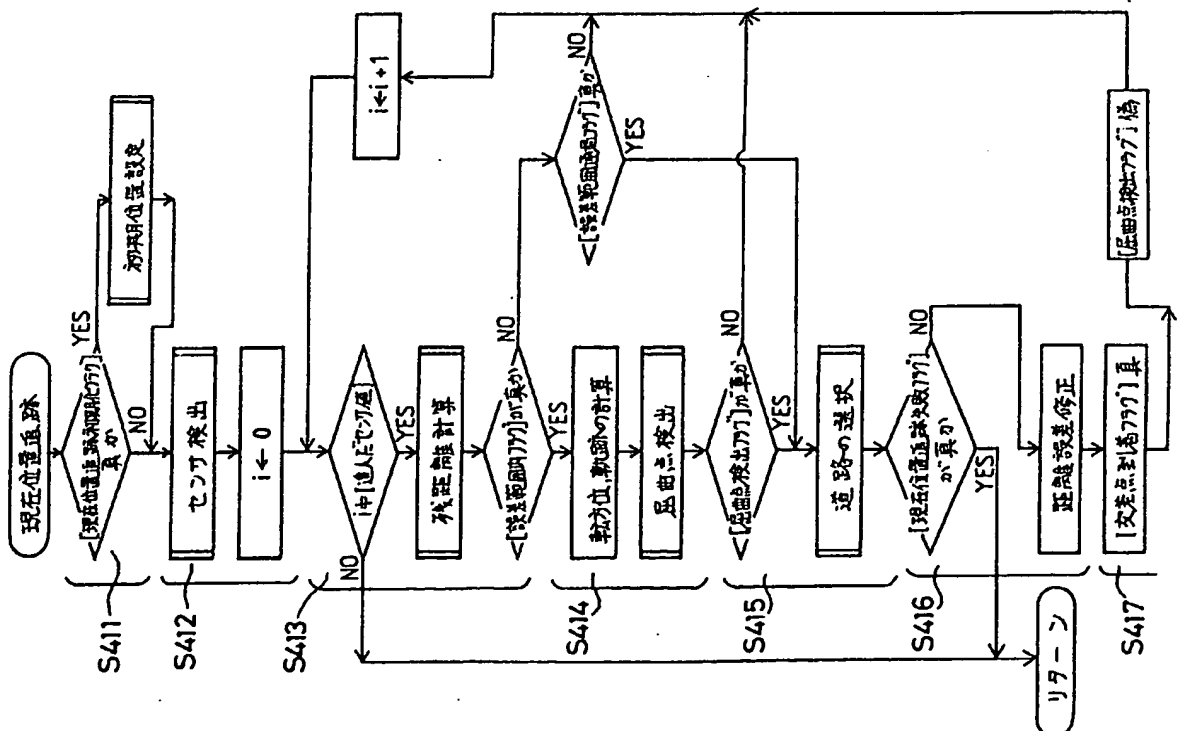
第10図



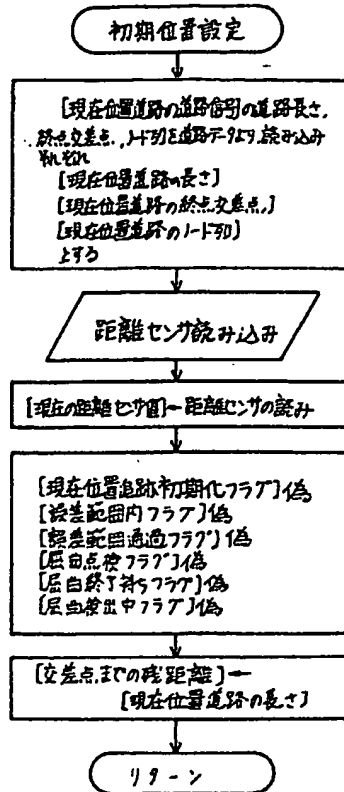
第9図



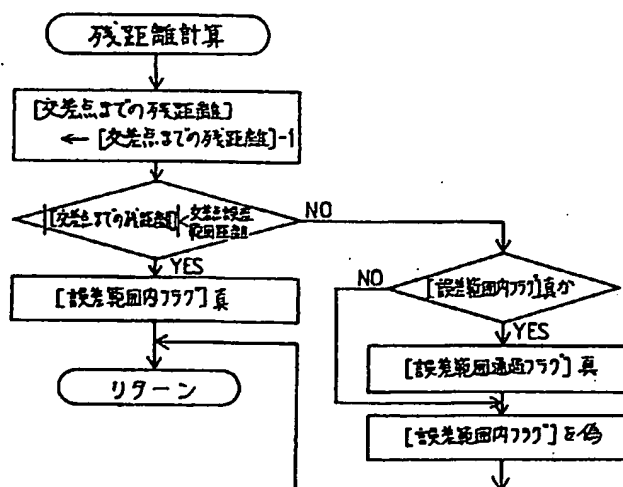
第11図



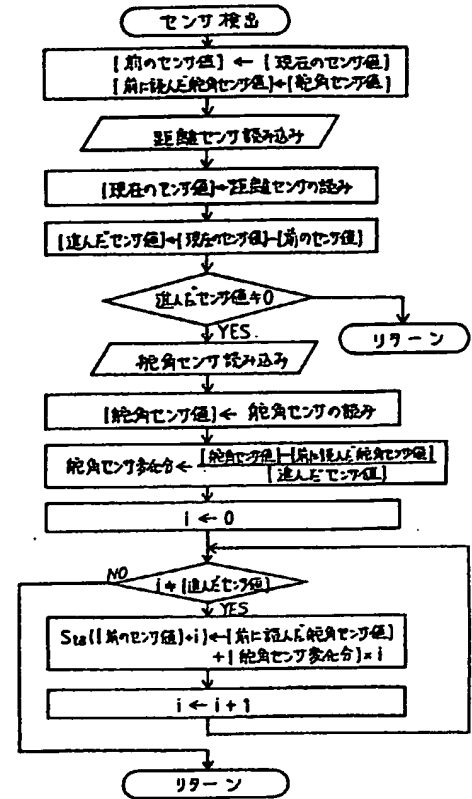
第12図



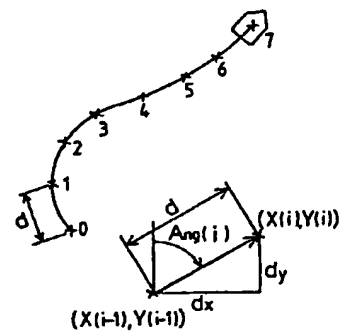
第14図



第13図



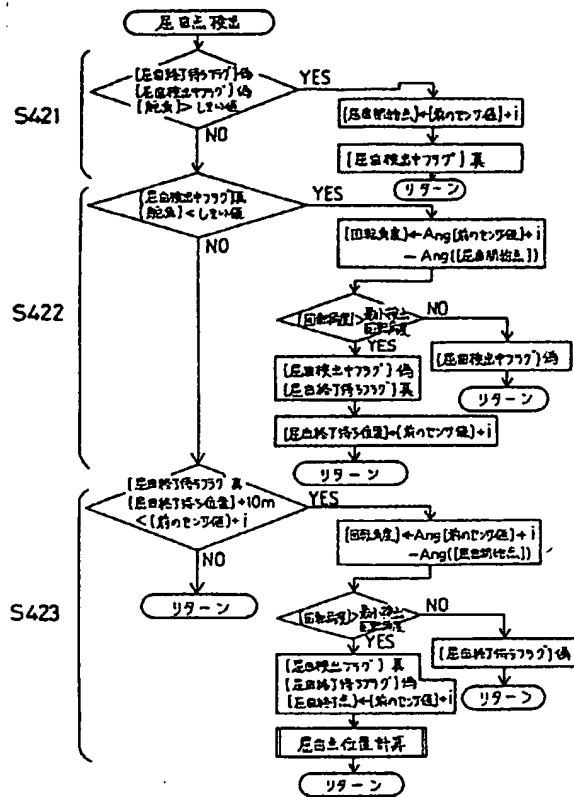
第15図



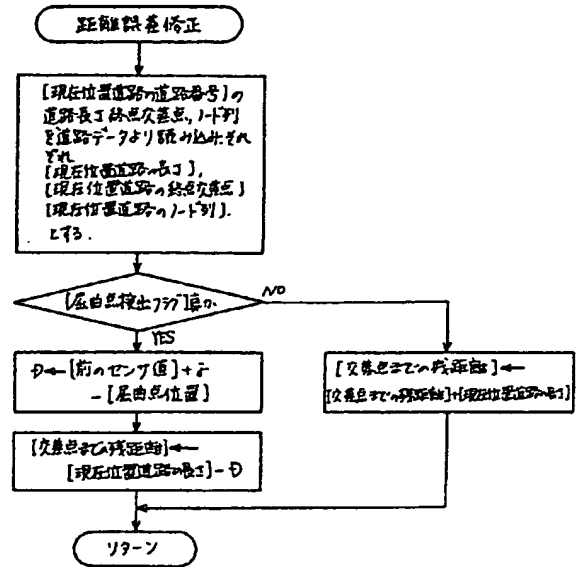
舵角Sta(i)	車両方位Ang(i)	車両軌跡(X(i), Y(i))
0	-4	0 (100, 100)
1	3	1 (99, 105)
2	0	
6	5	6 (150, 120)
7	4	7 (155, 125)



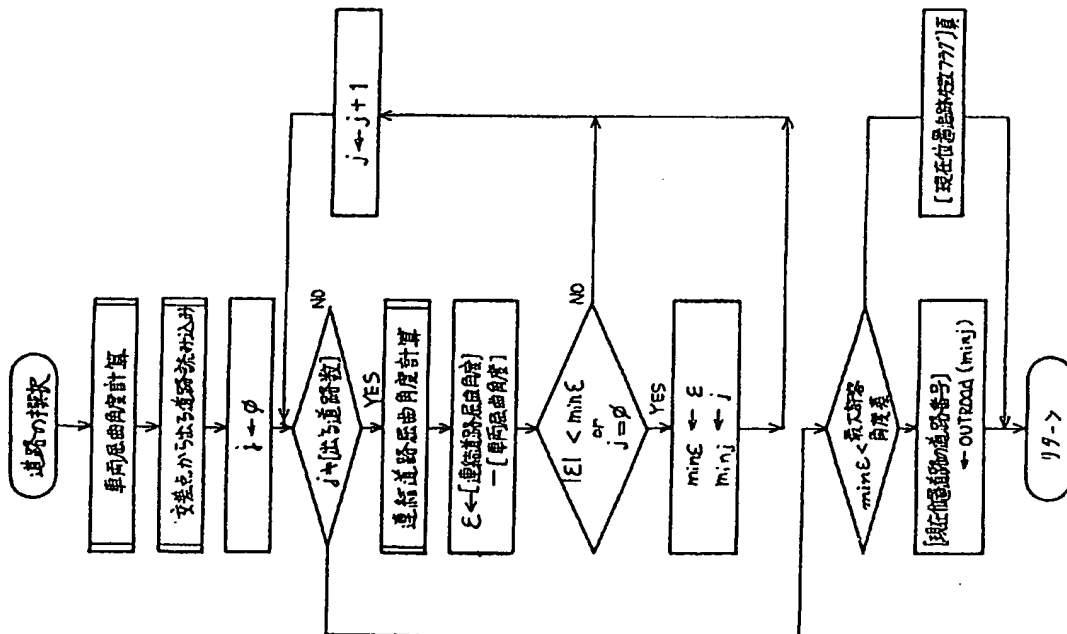
第16 図



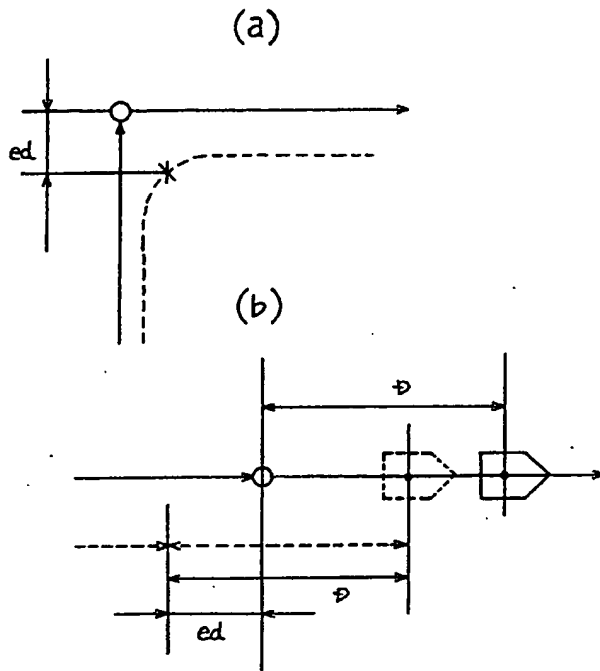
第18 図



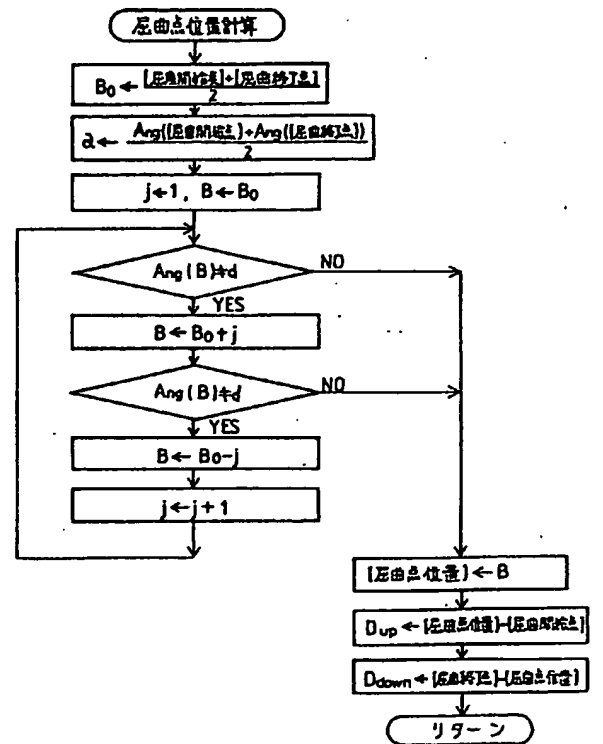
第17 図



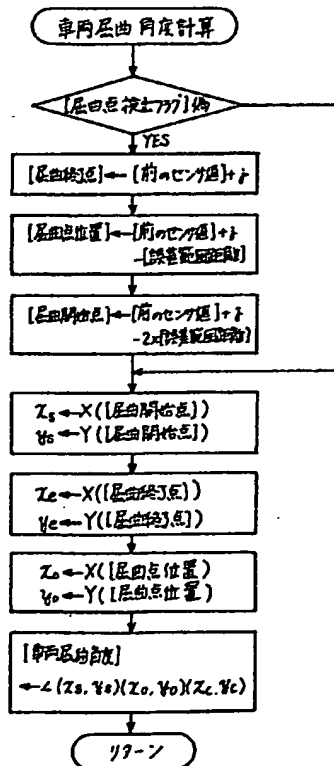
第19 図



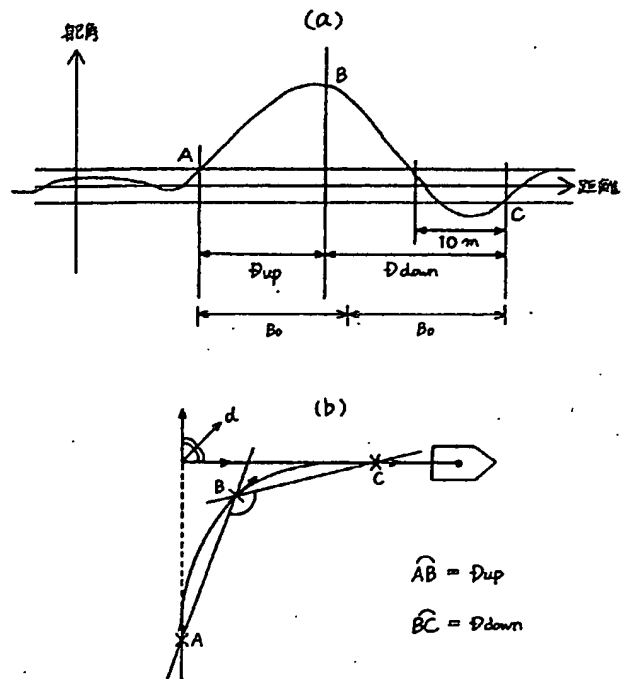
第20 図



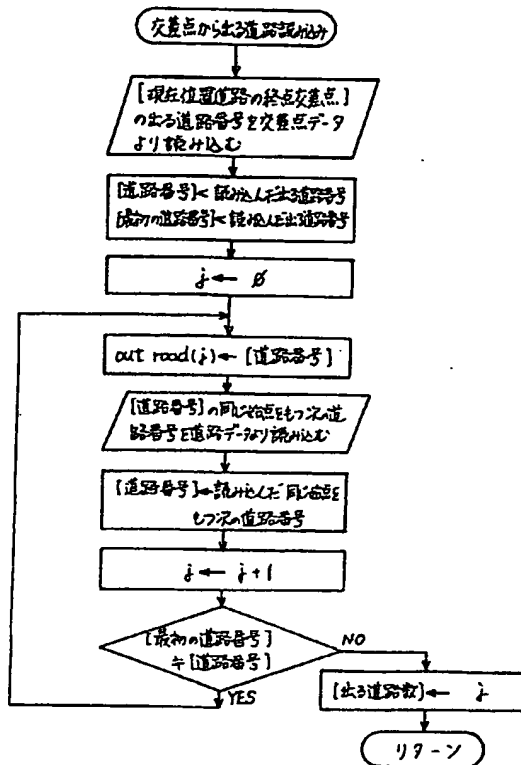
第21 図



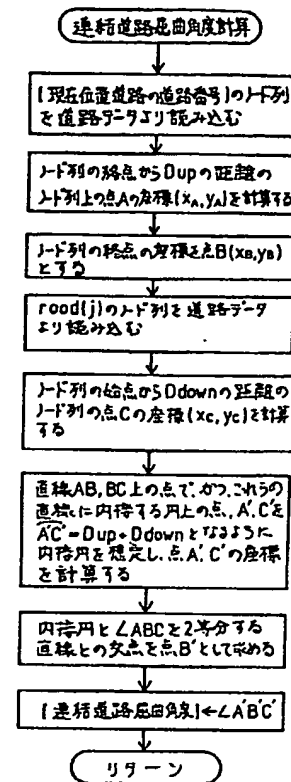
第22 図



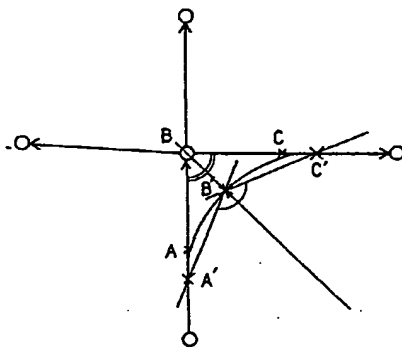
第23 図



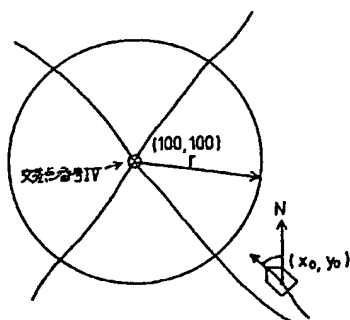
第24 圖



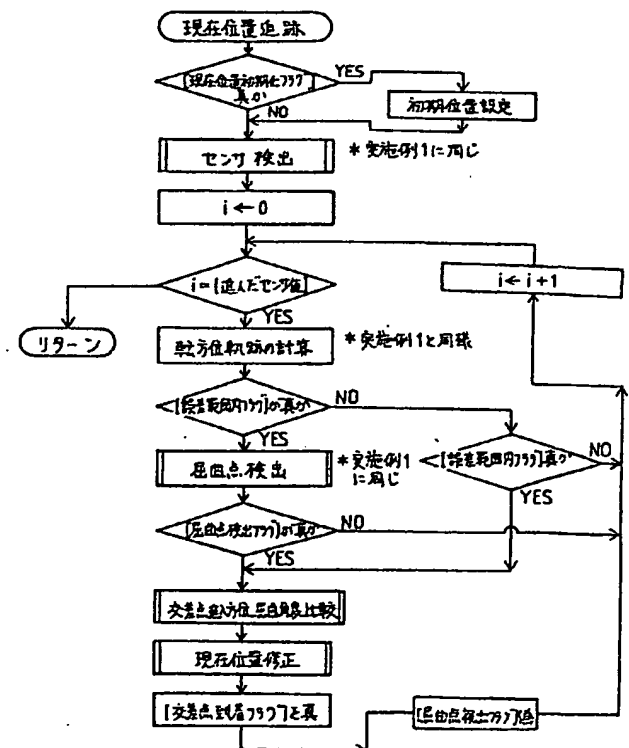
第25 図



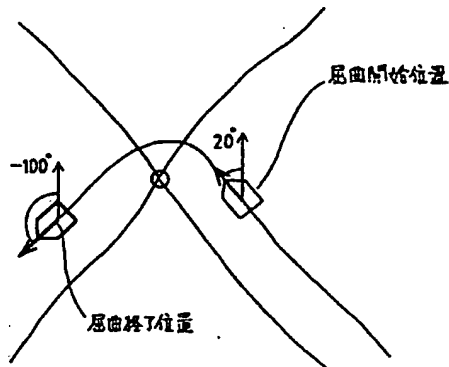
第27图(a)



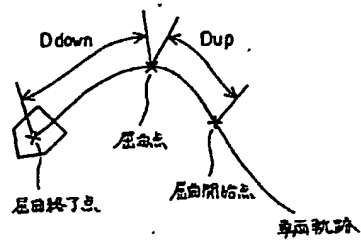
第26 図



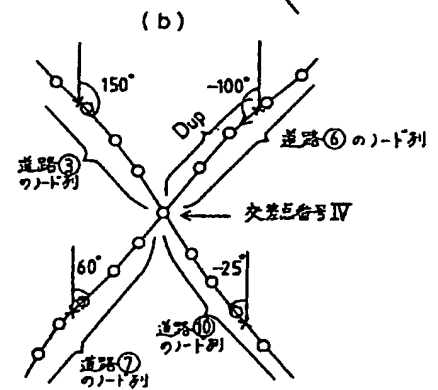
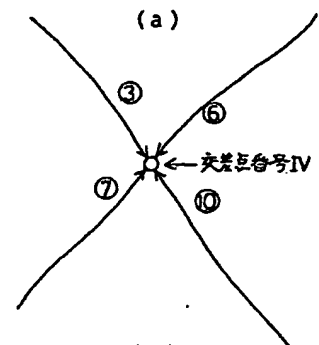
第27図 (b)



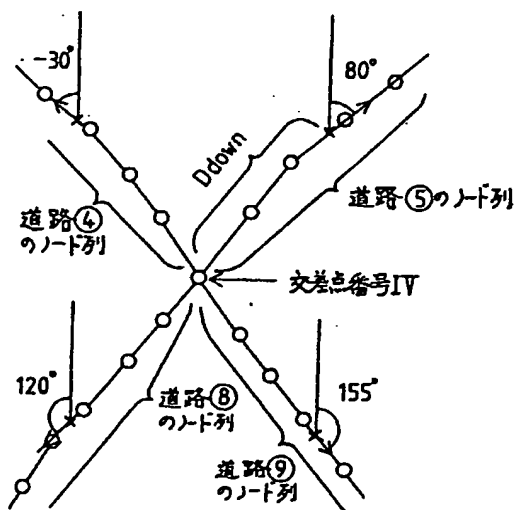
第27図 (c)



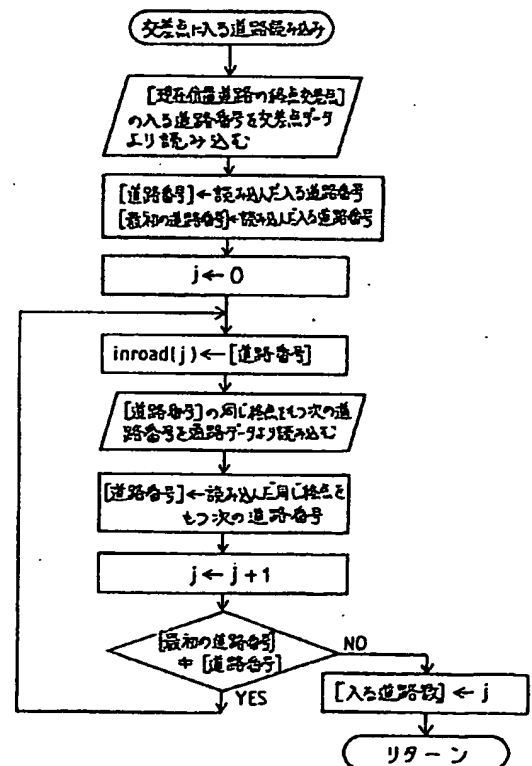
第28図



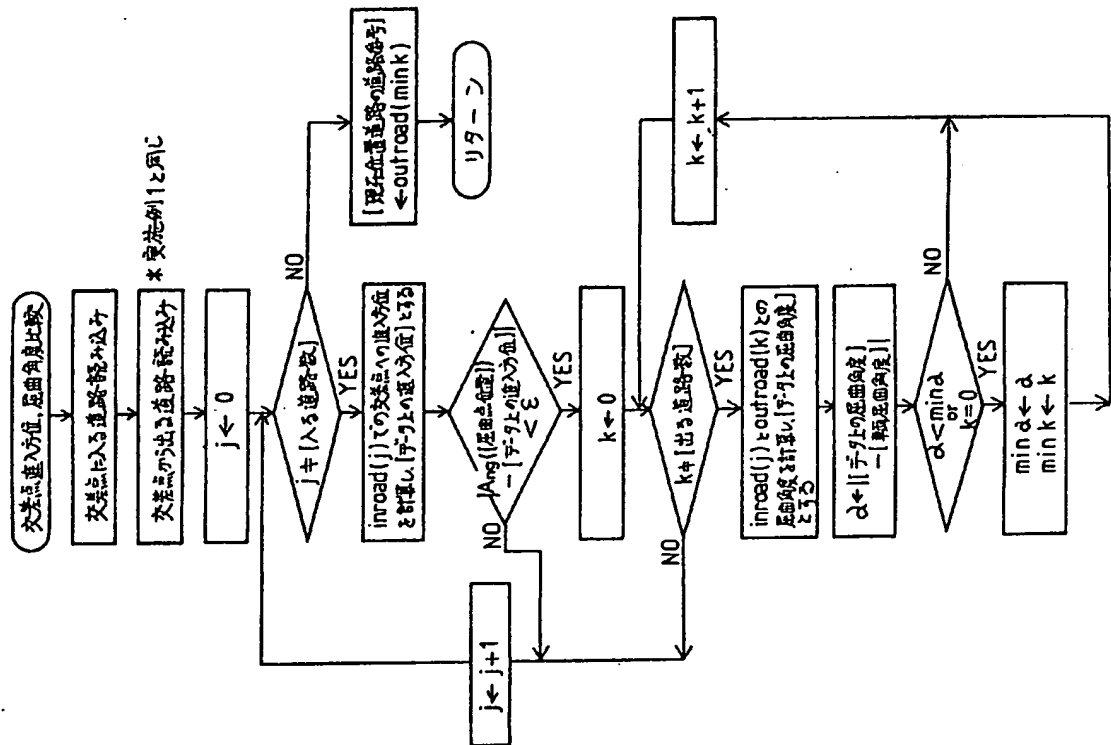
第28図 (c)



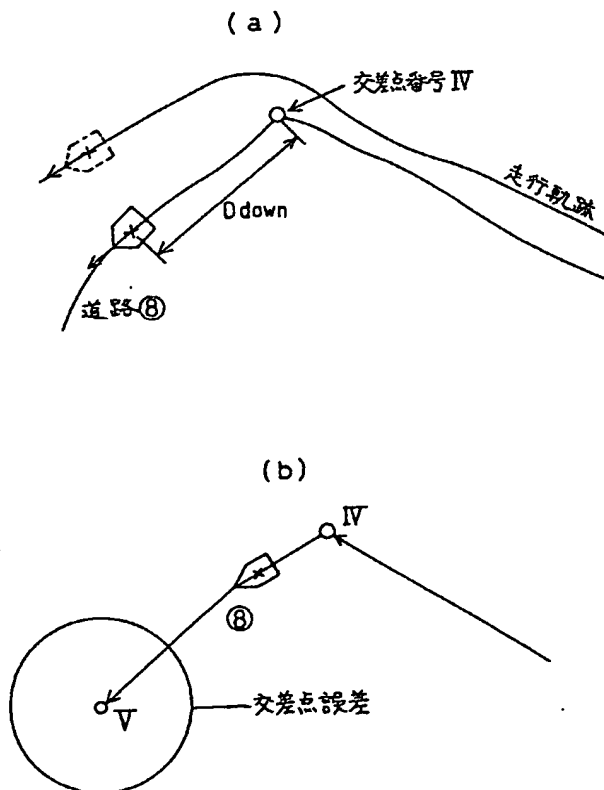
第30図



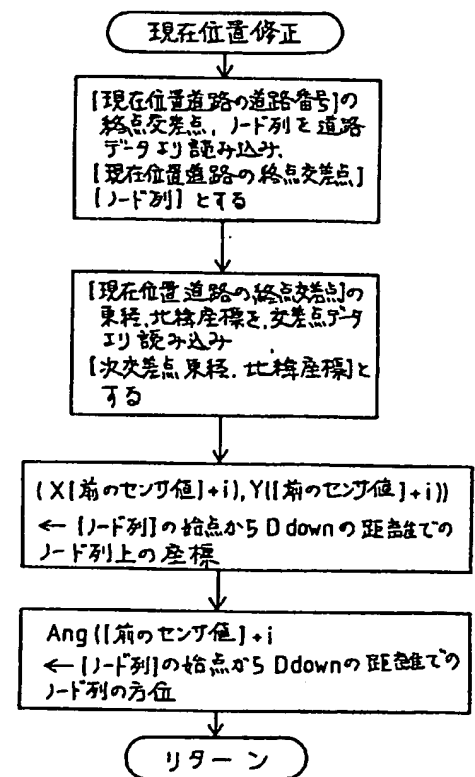
第29図



第31図

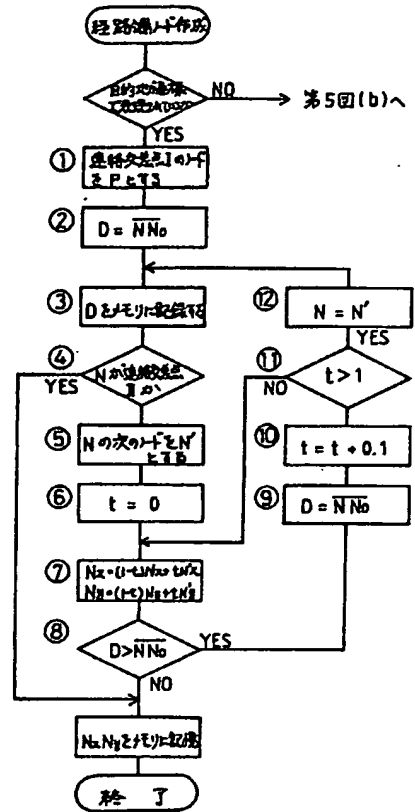
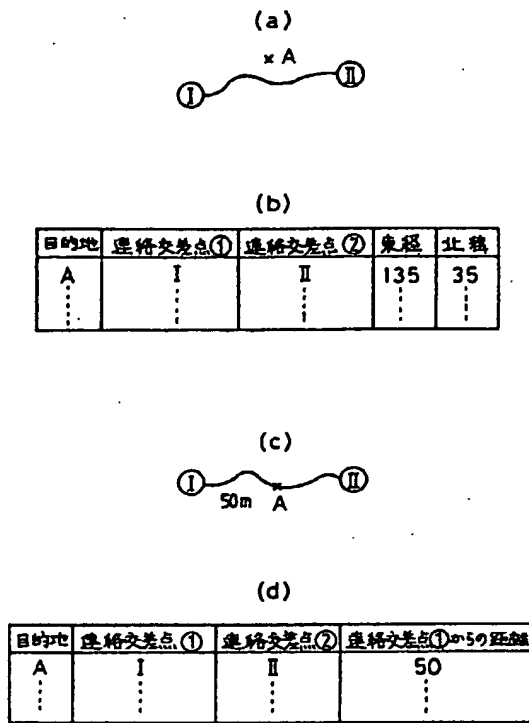


第32図

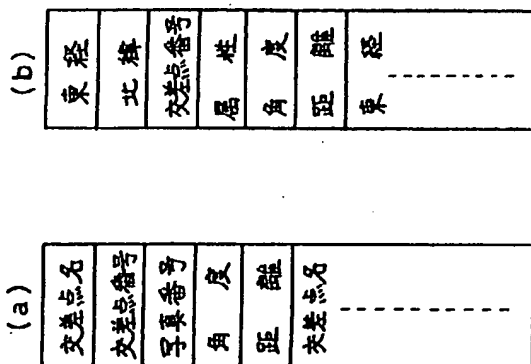


第35図(a)

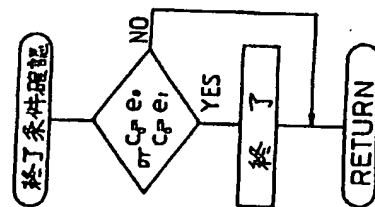
第33図



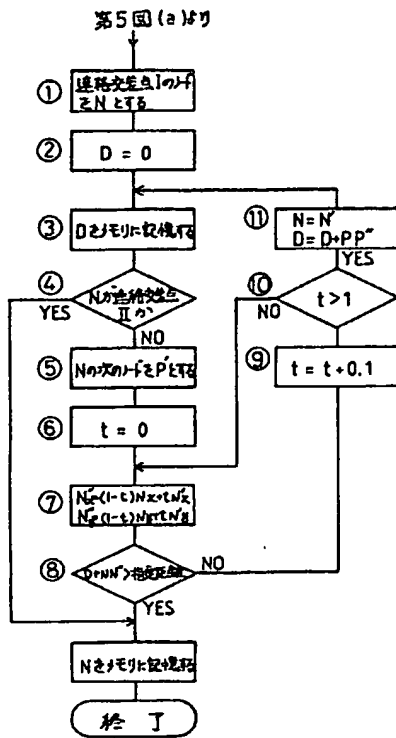
第34図



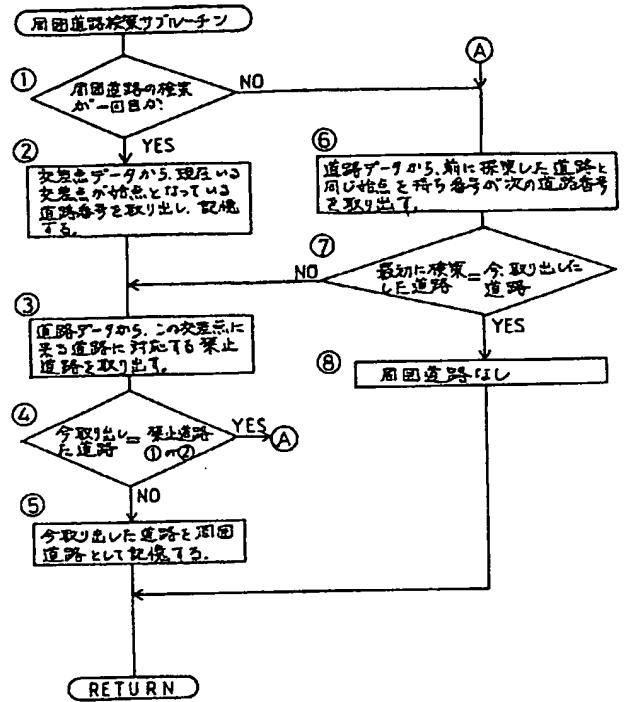
第39図



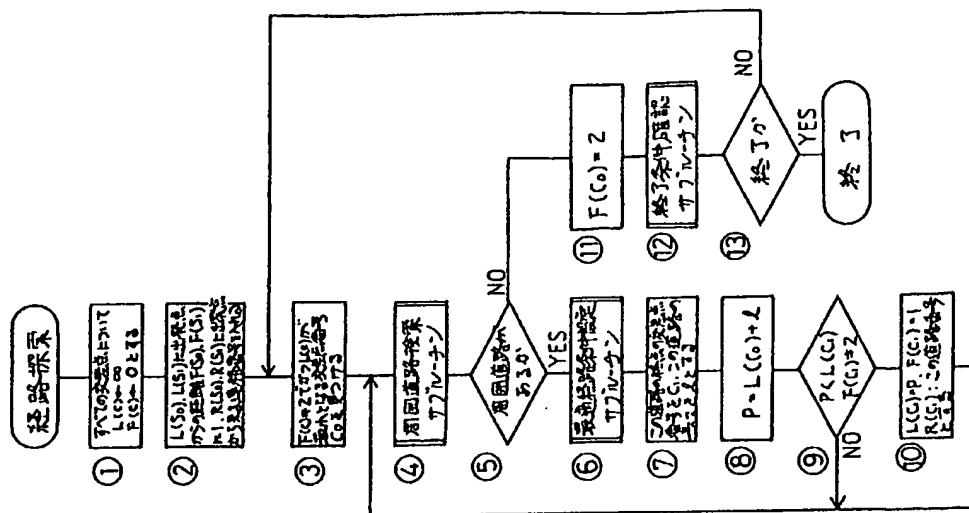
第35図(b)



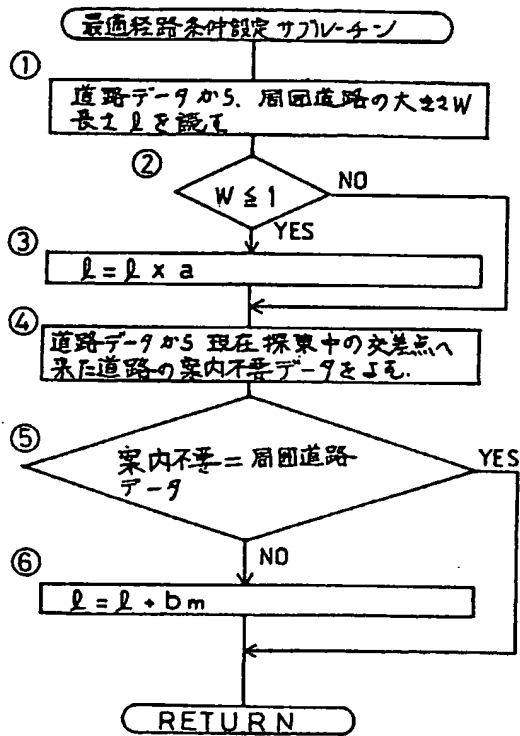
第37図



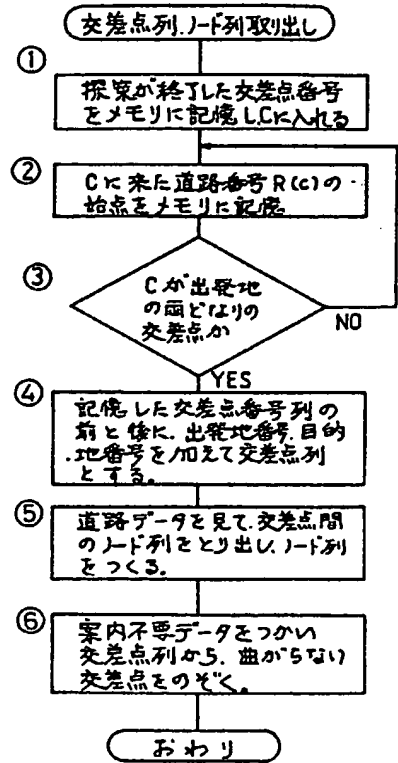
第36図



第38 図



第40 図



第1頁の続き

②発 明 者 横 山 昭 二

愛知県安城市藤井町高根10番地 アイシン・エイ・ダブリ  
ュ株式会社内



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**